

فيجول بيسك

تأليف

حيدر كاظم حمود

صادق عبد العزيز مهدي

الجامعة المستنصرية / كلية التربية

2016

الفهرست

الصفحة	الموضوع
	الفصل الاول : بيئة وداوات فيجول بيسك
	1-1 مقدمة
	2-1 بدء تشغيل فيجوال بيسك
	3-1 فتح مشروع في فيجوال بيسك :
	4-1 تشغيل البرنامج :
	5-1 إيقاف تنفيذ مشروع الفيجوال بيسك :
	6-1 إيقاف الفيجوال بيسك :
	7-1 بيئة فيجوال بيسك:
	8-1 أجزاء مشروع الفيجوال بيسك:
	9-1 أجزاء برنامج الفيجوال بيسك:
	10-1 بناء البرنامج الأول
	11-1 ضبط الخواص في طور التشغيل :
	12-1 كيفية استخدام أسماء الأدوات في إجراءات الأحداث :
	13-1 كتابة إجراء الحدث :
	14-1 حفظ مشروع الفيجوال بيسك :
	15-1 أداة زر الأوامر command button
	16-1 اداة العنوان Label Control :
	17-1 أداة النص Text Box Control
	18-1 الدرس الثاني في لغة البيسك :
	19-1 أشرطة التمرير والمنطق ورسائل الإدخال والإخراج Scroll Bar
	20-1 زر الاختيار Option Button :
	21-1 خانة التحقق Check Box :
	22 -1 اداة الصورة المحدودة Picturebox :

	23-1 اداة الصور Image Box
	24-1 القوائم Menue
	الفصل الثاني : المتغيرات والعمليات الرياضية
	1-2 المتغيرات Variables
	2-2 أنواع المتغيرات Variable Types
	3-2 العمليات الحسابية Arithmetic Operators
	4-2 دوال رياضية
	5-2 التعامل مع التاريخ
	6-2 السلاسل الحرفية
	7-2 صناديق الإدخال والإخراج
	8-2 مجموعة أمثلة
	9-2 مشروع - حساب الادخار
	الفصل الثالث : ادوات المنطق و الشرط و العدادات والمصفوفات
	1-3 المنطق والشرط
	2-3 الاداة If ..Then..Elseif ..Endif
	3-3 أداة الشرط أو OR :
	4-3 أداة النفي Not
	5-3 العدادات والدورات Loops
	6-3 دالة العدد العشوائي RND :
	7-3 مجموعة امثلة
	8-3 المصفوفات
	الفصل الرابع : إدارة قواعد البيانات
	1-4 مقدمة :
	2-4 الأداة Data :

	3-4 ضبط الأداة Data على قاعدة البيانات :
	4-4 طريقة برنامج أكسس :
	5-4 الأداة DataGridView
	6-4 التحكم في الأداة Data
	الفصل الخامس :الرسوم والصور والصوت
	1-5 ادوات الرسم
	2-5 رسم الخطوط
	3-5 رسم المستطيلات
	4-5 رسم الدوائر والاشكال البيضوية
	5-5 رسم النقاط
	6-5 الرسم باستخدام ادوات الرسم
	7-5 اهم احداث الفورم الخاصة بالرسوم
	8-5 امثلة
	الفصل السادس :الشبكات والانترنت
	1-6 مقدمة :
	2-6 حل شفرة الاسم Name Resolution :
	3-6 المقبس Socket :
	4-6 أداة الإنترنت Inet
	5-6 مشروع :
	6-6 الأداة WinSock :

	7-6 أداة مستعرض المواقع Web Browsing
	8-6 الأداة Scriptlet
	9-6 مشروع بناء متصفح
	10-6 لغة VB Script
	11-6 استخدام فيجوال بيسك إسكربت Visual Basic Script و قياسية المستعرض Explorer
	المصادر

الفصل الاول

بيئة وادوات فيجول بيسك

1-1 مقدمة

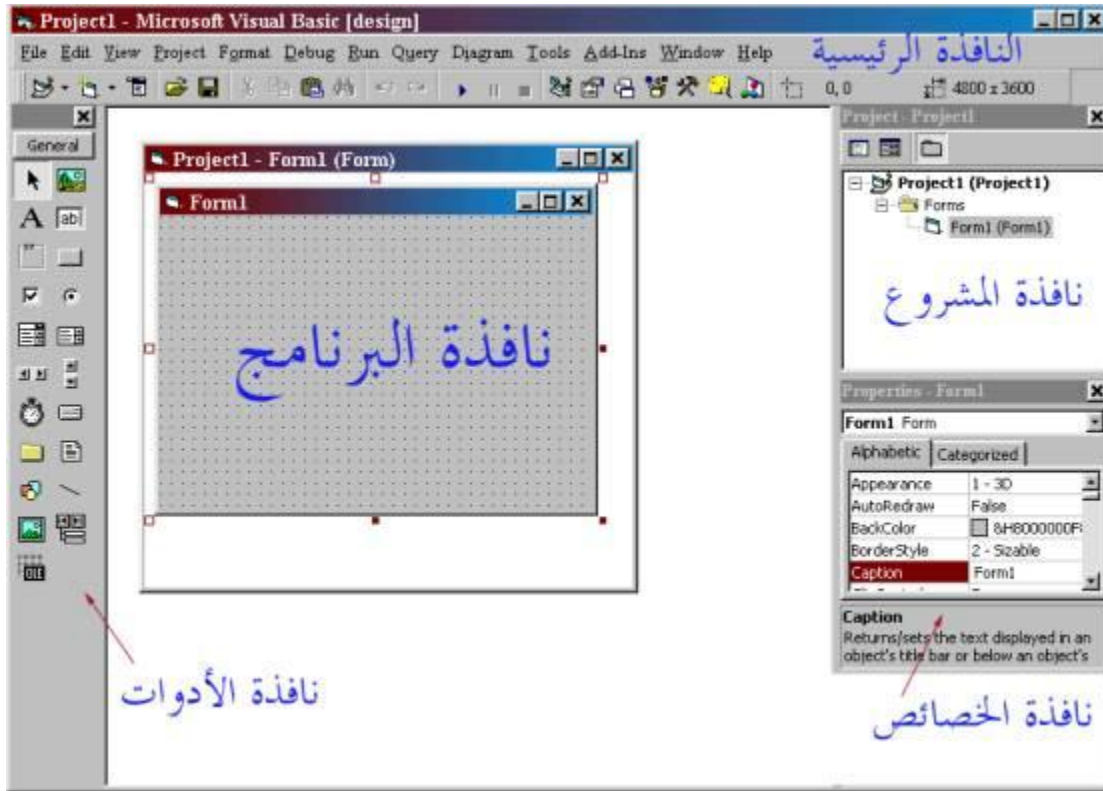
أنتجت شركة مايكروسوفت أول إصدار من لغة البيسك عام 1970م ، وسمي Basic والأسم يعتبر اختصار للكلمة باللغة البرمجة العامة التسلسلية للمبتدئين (Beginner's All-Purpose Symbolic InstructionCode) ، وتوالت الإصدارات فظهر الإصدارات GW-BASIC, QuickBasic and QBasic ، و كلهم يعملون في بيئة Dos ، و مع انتشار بيئة ويندوز ظهرت فيجوال بيسك التي احتوت على كثير من أوامر QBASIC وأضافت العديد من الوظائف التي جعلت من البرمجة بفيجوال بيسك يسره و سهلة .

على الرغم من سهولة فيجوال بيسك إلا أننا سوف نفترض أن لدينا بعض المهارات منها :

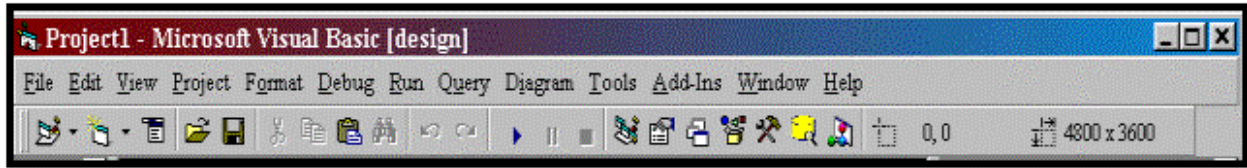
- 1- تشغيل الكمبيوتر و استخدام الماوس .
- 2- معرفة قدر من المعلومات عن بيئة ويندوز .
- 3- معرفة التحكم في النافذة بالماوس من حيث تصغيرها وتكبيرها ونقلها .
- 4- معرفة كيفية تشغيل التطبيق من خلال قائمة ابدأ .
- 5- الألفة بملأ النوافذ التي يجب أن تملأها بالمعلومات .
- 6- معرفة كيفية إنشاء المجلدات والبحث فيها عن ملفات بمختلف أنواعها سواء الملفات النصية أو الملفات التنفيذية أو ملفات الصور .

1-2 بدء تشغيل فيجوال بيسك

نحن نفترض أن لدينا الإصدار السادس من فيجوال بيسك 32 بت ، ننقر قائمة ابدأ و نختر خيار البرامج ثم ننقر قائمة فيجوال بيسك 6 فتظهر لنا النافذة التالية بعد أن نختر الطور القياسي .
أول شئ تقابله هو شريط العنوان و يقع في اعلى الشاشة و هو يقدم معلومات تفيد بأنك تصمم برنامج باستخدام فيجوال بيسك ، وتحت شريط العنوان نجد شريط قائمة فيجوال بيسك حيث يمكننا التحكم في مشررك ، ويجب أن تألف العمل مع القوائم ، فقوائم فيجوال بيسك شبيهه بقوائم ببرنامج معالجة الكلمات ورد ، ويلي شريط القائمة الرئيسية شريط الأدوات الذي يحوي بعض الأدوات المفيدة والتي سوف توفر لنا بعض وظائف فيجوال بيسك بطريقة سهلة وسريعة .



وشريط الأدوات يحوي صور وكل صورة تؤدي وظيفة محددة ، وإذا نحاول تمرير الماوس فوق أي من تلك الصور فسوف تحصل على رسالة تفيد بوظيفة ذلك الزر .



وان فتح مشروع يكون:



فالقائمة التي تنبثق بمرور الماوس على الصورة توفر مساعدة سريعة حول الزر .
ويحتوي شريط الأدوات على مجموعة الأوامر التي تستخدمها دائما في فيجول بيسك هذه الأوامر موجودة في شريط القوائم ولكنها وضعت هنا تسهيلا عليك. مثلا بدل ما تبحث عن امر فتح ملف من قائمة File يمكننا ان ننقر على الرمز مباشرة

وشرح للرموز من اليسار الى اليمين : كالتالي

مشروع جديد

نافذه جديده

اظهار نافذة تصميم القوائم

فتح مشروع

حفظ مشروع

تشغيل البرنامج الحالي يعني المشروع الحالي

إيقاف البرنامج بصورة مؤقتة

إنهاء البرنامج الحالي

إظهار نافذة الخصائص

وضع أو حذف نقطة توقف

مراقبة متغير

عرض الاستدعاءات

تنفيذ البرنامج خطوة بخطوة

تنفيذ البرنامج إجراء فإجراء

3-1 فتح مشروع في فيجوال بيسك :

كلمة مشروع تشير لبرنامج معد بلغة فيجوال بيسك ، فالمشروع يحوي معلومات هامة للكمبيوتر تحدد له ما سوف يفعله عند تنفيذ البرنامج وعند فتح المشروع سوف ننقر على زر فتح المشروع التمس صورته هي :



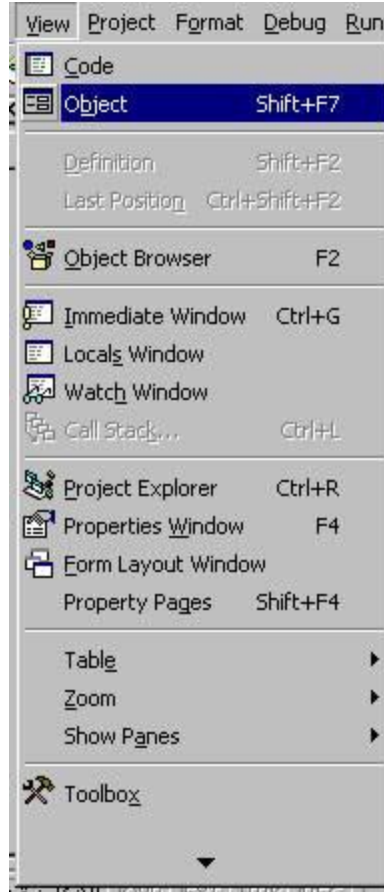
□ ننقر هذا الزر ، وهنا سوف تظهر نافذة فتح مشروع.

□ حمل البرنامج Sample .

□ نشغل البرنامج وسوف نجد نافذة تحوي عدة نوافذ جزئية .



ننقر قائمة View من القائمة الرئيسية ثم نختار هدف Object



ويمكننا أن نختار object مباشرة بلمس المفاتيح **Shift + F7** معا من الواضح أن لغة فيجوال بيسك تحوي العديد من الأوامر والإجراءات وبممارسة اللغة أفضل وسيلة لتذكرها.

وتسمى نافذة المشروع **Form** ، وكل المشروعات التي سوف تنفذها في فيجوال بيسك نحتاج لنافذة ، وفي الواقع أن كل نافذة مشروع له عنوان يظهر على شكل شريط أعلى النافذة مع أيقونة ، والعنوان يصف ما سوف ينفذه المشروع كما أن لدينا ثلاثة أزرار صغيرة تتحكم في نافذة المشروع ، ويوجد أشياء أخرى على نافذة المشروع ، وهي تعتبر أساسية في فيجوال بيسك.

في الشكل السابق، كما نرى أزرار مربعة تحمل اسماء العاب، كما نرى صور الألعاب ، ونرى أيضاً أزرار - دوائر- تحمل أسماء الألوان ، كما لدينا شريط يسمى شريط تمرير، ولديك زر كبير يحمل الاسم صوت ولديك كرة في مستطيل مع زر اسمه ابدأ ، وهو يشبه ساعة إيقاف ، وكل من تلك الأشياء تسمى أدوات تحكم **Controls** أو أغراض **objects** ، وتلك الأدوات تهبيئ قناة **interface** أو خط اتصال بينك و بين الكمبيوتر لتخبره عما نريد فعله ، ويقوم الكمبيوتر بتنفيذ ما أشرت به ، ومن المفيد أن نتعرف أن

نافذة المشروع نفسه أداة تحكم ، وإذا كنت قد سبق لنا أن استخدمت تطبيقات ويندوز من قبل ، فقد تكون استخدمت أدوات مشابهة مثل الأزرار وأشرطة التمرير وغيرها .

والواقع أن أدوات التحكم من أهم أساسيات فيجوال بيسك ، فهي لبنات تساعدك في بناء مشروعنا ، ولكن لنأخذ بعض الخبرة عن طريقة تشغيل برامج فيجوال بيسك من خلال تنفيذ البرنامج Sample وقبل ذلك ننظر الى شريط فيجوال بيسك وسوف نجد كلمة [design] محاطة بقوس مربع ، وهذا يخبرنا بأن طور فيجوال بيسك هو التصميم ، وهذا الطور يستخدم في بناء مشروع فيجوال بيسك ، ننظر لهذا الشريط لت نلاحظ الطور الذي تعمل فيه فيجوال بيسك (فلدى ك أيضاً طور التنفيذ أو التشغيل run).

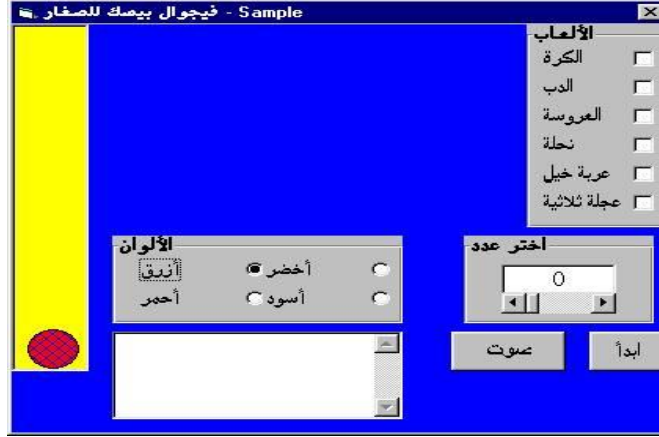
4-1 تشغيل البرنامج :

بعد بناء المشروع بلغة فيجوال بيسك نحتاج لتنفيذ البرنامج ، و يتيح للمستخدم بدء التفاعل مع البرنامج، ومن ثم يقوم الكمبيوتر بتنفيذ مهمة معينة ، ويمكننا أن ننفذ البرنامج أو نشغله باستخدام شريط الأدوات ، ابحث في شريط الأدوات عن زر يشبه زر تشغيل مسجل الفيديو كاسيت أو زر تشغيل مسجل الصوت إنه يشبه الشكل التالي :



□ ننقر الزر السابق لتشغيل البرنامج Sample الذي تم تحميله سابقاً

حينئذ سوف يظهر شكل نافذة المشروع كما يلي :



نلاحظ أن بعض الأشياء قد تغيرت ، فكل صور الألعاب قد اختفت ، كما أن لون خلفية نافذة المشروع أصبح أزرق ، كما أن الدائرة التي تشير للون الأزرق به دائرة سوداء صغيرة ، كما أن أيقونة ساعة التوقيت قد اختفت، كما أن الكرة تظهر في أسفل المستطيل الأصفر .

سنعرف كيف تم ذلك لاحقاً وكلما تقدمنا في دراسة فيجوال بيسك، كما نلاحظ ان النافذة الرئيسية تغير طور فيجوال بيسك الى طور التشغيل run .

ورغم ان مشروع البرنامج في طور التشغيل ، ولكن لا شيء يتم تنفيذ فيجوال بيسك تنتظر لحدث **event** سوف تقوم به أنت ، ونذكر أن فيجوال بيسك لا تنفذ شيء حتي تقوم بالحدث الذي تم وصفه في البرنامج ، ولذلك تسمى فيجوال بيسك بأنها لغة موجهة بالأحداث.

الحدث يحدث عندما تفعل شيء ما على نافذة المشروع مثل النقر بالماوس أو كتابة كلمات في أماكن خاصة أو سحب هدف على نافذة المشروع ، ففي الركن الأيمن العلوي من نافذة المشروع توجد 6 مربعات داخل منطقة مستطيلة عنوانها **الألعاب** وكل مربع صغير يحمل اسم لعبة **نقر** واحد منها و نلاحظ ماذا يحدث على نافذة المشروع العلامة - x - التي تظهر بالمربع تشير الى اننا نختار ذلك المربع ، ومن ثم يظهر شكل اللعبة التي يحملها المربع على نافذة المشروع .

والكمبيوتر يعرف الحدث - من خلال البرنامج - والحدث يخبر الكمبيوتر أن يظهر هنا شكل اللعبة ، ونقر المربع مرة أخرى يخفي اللعبة وكذلك علامة الاختيار تختفي ، والواقع اننا أحدثت حدث آخر أخبر الكمبيوتر أن يخفي اللعبة ، و الأداة التي تفعل ذلك تسمى **مربع الاختيار check box** ، و نلاحظ انه يمكننا أن نختار عدة مربعات اختيار دفعة واحدة ، وكلما نختار احد مربعات الاختيار تظهر الصورة المصاحبة له .

وفي المثال الحالي لدينا أيضاً أداتين ، فالمنطقة المستطيلة التي تحوي مربعات الاختيار تسمى أداة الإطار **frame** والمنطقة التي تظهر بها صور الألعاب تسمى بأداة الصورة .

وفي المنتصف تقريباً لدينا إطار - حاوية - بها أزار عبارة دوائر بيضاء و عنوان ذلك الإطار الألوان وكل زر اختيار يحمل لون معين ، فالزر الذي يحمل اسم اللون الأزرق نجد به دائرة سوداء ، وتلك النقطة أو الدائرة السوداء تشير لاختيار ذلك الزر ، وهنا نلاحظ أن لون نافذة المشروع قد أصبح أزرق - نحاول ان نختار زر اختيار آخر و نلاحظ الفرق - هل يمكننا أن نختار زررين معا أو كلهم مثل مربعات الاختيار ؟

وحدث النقر يخبر الكمبيوتر أن يغير لون خلفية نافذة المشروع ، واختيار لون آخر يخفي النقطة من زر الاختيار السابق ، وهذا هو الفرق بين مربع الاختيار و زر الاختيار ، فيمكننا أن نختار زر اختيار واحد ، بينما يمكننا أن نختار عدة مربعات اختيار دفعة واحدة .

وتحت إطار الألعاب نجد اطار الألعاب يوجد إطار آخر عنوانه **نختار عدد** ، ونرى عدد يظهر في اداة تحكم تسمى أداة عنوان **label**- مرر الماوس على الشكل لتتعرف اسمه -وتحت العنوان أداة أخرى تسمى شريط تمرير **scroll bar** ، ولعلك شاهدت شريط التمرير في أحد تطبيقات ويندوز ، وفي البرنامج الحالي يستخدم شريط التمرير في تغيير العدد الموجود في أداة العنوان و نقر شريط التمرير يزيد العدد الموجود في أداة العنوان بواحد ، والنقر على شريط التمرير نفسه يزيد العدد بعشرة ، وسحب مؤشر شريط التمرير يغير العدد ايضاً ، وأكبر قيمه هي 100 و أصغر قيمة هي صفر .

ونقر شريط التمرير يجعل الكمبيوتر يستجيب لحدث التغير **change** الذي يحدث في كل مرة يتحرك فيها مؤشر شريط التمرير ، وحدث التغير يخبر الكمبيوتر ان قيمة شريط التمرير - موقع المؤشر - قد تغير ، ومن ثم تظهر القيمة الجديدة في أداة العنوان - ننقر اى جزء من شريط التمرير و نضغط مفتاح السهم يمين ماذا نلاحظ ؟

وتحت إطار اختيار العدد توجد منطقة بها شريط تمرير تسمى بأداة النص **text box** ، ويمكننا نقرأها وكتابة أي نص بها - جرب أن نختار الأداة بالمفتاح Tab ، وأداة النص تشبه برنامج معالجة الكلمات ، ففي كل مرة تكتب فيها في أداة النص تحدث احداث عدة فلدى ك حدث ضغط مفتاح وحدث تغير يتم كلما تغيرت الكتابة الموجودة في خانة النص أو أداة النص .

وبالقرب من أداة النص نرى زر أوامر **command button** يحمل الأسم صوت ، ونقر الزر يؤدي لسماحك صوت بييب من سماعة الزر الداخلي ، و أزرار الأوامر من أكثر الأدوات التي تستخدم في فيجوال بيسك .

وآخر شئ على نافذة المشروع هو المستطيل الأصفر وهو عبارة عن أداة الصورة **picture box** وهو يحوي دائرة صغيرة تسمى أداة الشكل **shape** ، وتحت الصورة لدينا زر يحمل العنوان ابدأ ، ونقر ذلك الزر - هذا حدث نقر - وهنا تبدأ الكرة تتحرك لأعلى وتستمر في التحرك لأعلى حتي تلمس حافة

المستطيل العليا والدنيا و هنا يتغير عنوان الزر ابدأ الى توقف و هنا نقر الزر يوقف حركة الكرة ، و الآن لنتذكر رمز ساعة الإيقاف التي كانت تظهر في طور التصميم واختفت في طور التشغيل ، في فهي تستخدم في التحكم في قفزة الكرة ، وتلك الأداة تسمى اداة التوقيت **timer** فنقر الزر عندما يكون عنوانه ابدأ ينقل التحكم لأداة التوقيت، ونقره عندما يكون عنوانه توقف يوقف عمل أداة التوقيت، وأداة التوقيت أو الساعة تولد أحداث زمنية كل 1\10 من الثانية ، وأثناء ذلك يتغير موضع الكرة مما يظهر حركة الكرة ، وأثناء ذلك يمكننا ان تظهر صور الألعاب ، أو الكتابة او تغيير الألوان أو إصدار الصوت ، أي أن حدث التوقيت يعمل بمفرده ، وهذا يعني ان فيجوال بيسك لديها مقدره على معالجة احداث عدة في نفس الوقت .
وكما نرى فالمثال الحالي هو مثال توضيحي لما يمكن ان تفعله فيجوال بيسك ، والمهم أن تلاحظ وأن تفهم ما يمكن فعله بمشروعات الفيجوال بيسك .
والمشروع يصنع من أدوات تتيح للمستخدم أن يزود الكمبيوتر بالمعلومات من خلال احداث ، ومن ثم يخرج الكمبيوتر النتائج المطلوبة .

5-1 إيقاف تنفيذ مشروع الفيجوال بيسك :

توجد عدة طرق لإيقاف تنفيذ مشروع الفيجوال بيسك ، وسوف نستخدم شريط الأدوات لإيقاف تنفيذ مشروع الفيجوال بيسك ، ابحث عن زر يشبه زر إيقاف الفيديو كاسيت أو مشغل الأقراص المدمجة ، أو مسجل الصوت ، وهو يشبه الزر التالي :



- إن النقر على ذلك الزر ينهي طور التنفيذ ويعود بك الى طور التصميم .
- ويمكننا ان تغلق البرنامج أو المشروع بنقر زر الإيقاف على نافذة المشروع وعلنه علامة × الى يمين شريط عنوان نافذة المشروع

زر الإيقاف

Sample - فيجوال بيسك للصغار

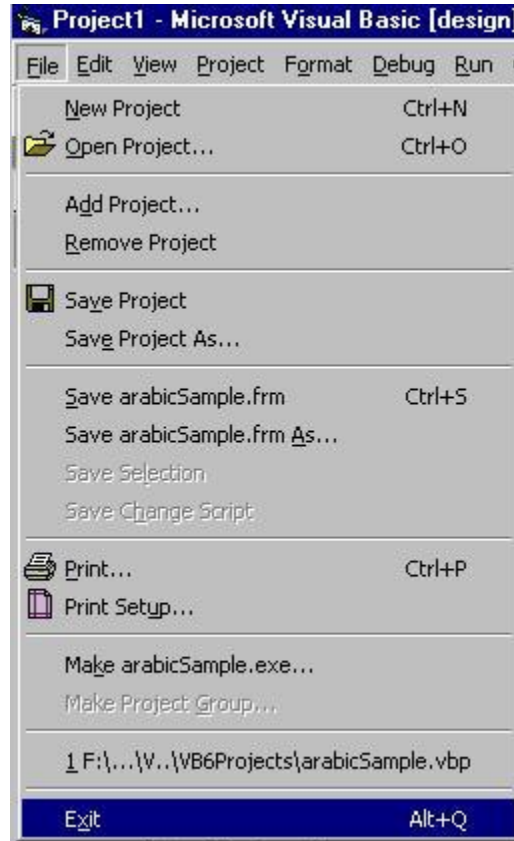
زر التعلق :



وزر التعلق وز الإيقاف السابق يختلفان فالأول يتيح إعادة استئناف البرنامج من النقطة التي توقفت عندها ، ومن ثم يمكننا مراقبة تغيير المتغيرات لضبط البرنامج ، بينما الثاني يخرج بك تماما من البرنامج .

6-1 إيقاف الفيچوال بيسك :

عندما تفرغ من تنفيذ مشروع الفيچوال بيسك ، فقد ترغب في ترك المشروع وربما ترك الفيچوال بيسك نفسه ، ولترك الفيچوال بيسك نتبع أحد إجرائين :



نختار قائمة ملف File من قائمة الفيچوال بيسك الرئيسية .

نختار الخيار خروج Exit وهو آخر خيارات تلك القائمة .

الآن توقف أو أغلق برنامج الفيچوال بيسك ، ومن ثم تغلق الفيچوال بيسك كل النوافذ المفتوحة ، و يتم إغلاق البرنامج Sample والخروج من فيچوال بيسك ، وسوف تطلب منك الفيچوال بيسك حفظ التغييرات التي أجريتها على البرنامج النشط ، والإجابة بنعم تحفظ التغييرات ، والإجابة بلا تخرج بدون حفظ للتغييرات .

والطريقة الأخرى لإغلاق الفيجوال بيسك هي بنقر زر الإغلاق الموجود على يمين شريط عنوان الفيجوال بيسك كما هو موضح بالشكل التالي :



والإجراء الأول موجود في معظم تطبيقات ويندوز ، والثاني أسهل منه بالطبع لأنه من خطوة واحدة ومباشر.

7-1 بيئة فيجوال بيسك:

في الفقرات السابقة، شرحنا شئ هام عن المشروع **project** ، وراينا أن المشروع يبنى على نافذة تسمى نافذة المشروع **form** وباستخدام أدوات ، ويتم التفاعل مع تلك الأدوات باستخدام أحداث **events** ، ومن ثم ينفذ الكمبيوتر مهام معينة طبقاً لتلك الأحداث ، وفي الفقرات القادمة سوف نتطرق الى خطوات بناء مشروعات الفيجوال بيسك بتأمل أجزاء المشروع ، وبالطبع سوف نتعرض لمصطلحات ومهارات جديدة .

8-1 أجزاء مشروع الفيجوال بيسك:

ذكرنا سابقا ان مشروع الفيجوال بيسك يتكون من ثلاثة أجزاء : شفرة المشروع ، و نافذة المشروع ، وأدوات تنفيذ المشروع . وكلمة مشروع تصف كل شئ نفعله حتى نحصل على ناتج من الكمبيوتر ، وتستخدم كلمات أخرى لوصف المشروع مثل تطبيق **application** أو برنامج **program** ، و نافذة المشروع هي نافذة يوضع عليها أدوات **Controls** تعد بمثابة قنوات اتصال مع الكمبيوتر ، والأدوات عبارة عن صور أو أدوات تو نضع على نوافذ المشروعات تتيح للمستخدم التفاعل مع الكمبيوتر ومن أمثلة الأدوات : مربعات النص، ومربعات العنوان، وأشرطة الأدوات و أزرار الأوامر، و نذكر أن نافذة المشروع يشار إليها بهدف **objects** .

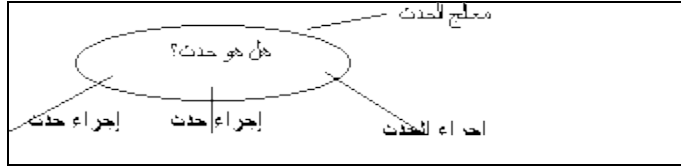
ومن ثم فالمشروع يتكون من نافذة مشروع تحوي عدة أدوات، وإذا فحصت ملفات الفيجوال بيسك فسوف نجد ملفين هما ملف المشروع وامتداده **vbp** وهو يعني مشروع فيجوال بيسك - وأحيانا **vbw** - و ملف نافذة المشروع وامتداده **frm** - أحيانا يكون الامتداد **frx** ويمكننا أن نجد أن البرنامج المذكور في

الحصة السابقة يتكون من ملفين هما : **Sample.vbp** و **Sample.frm** وأنت نحتاج لمعرفة ذلك النوعين من الملفات للتأكد من حفظ و صلاحية الملفات للعمل .

وثمة مفهوم يرتبط بمشروع الفيجوال بيسك وهو ما يعرف بالخاصية **property** ، فكل أداة لها خصائص - بما في ذلك نافذة المشروع- ، فمثلاً تتضمن الخواص عنوان الأداة **caption** ، واسمها **name** وحجمها **size** واللون **color** ، والخلفية ... **background** الخ .

كما ذكرنا ان لغة الفيجوال بيسك هي لغة موجهة بالأحداث **event-driven** أو أنها محكومة بمعالج للحدث **event processor** ، وهذا يعني أنه لا شئ يحدث في مشروع الفيجوال بيسك حتى تحدث بعض الأحداث على الأدوات .

وحيثما يكتشف الكمبيوتر وقوع حدث ما ، فإنه يجد سلسلة من التعليمات مرتبطة بهذا الحدث موصوفة في شفرة المشروع وهذا يسمى بإجراء حدث **event procedure** ، وعندما ينفذ الإجراء ، فإن تحكم البرنامج يعود للحدث :



إجراءات الأحداث تكتب في شفرة البرنامج وتوضع في الملف الذي امتداده **frm** ، وهو الملف الذي يحوي جمل الفيجوال بيسك .

9-1 أجزاء برنامج الفيجوال بيسك:

أن لغة الفيجوال بيسك أكثر من مجرد لغة برمجة فهي بيئة بناء مشروع ، وداخل تلك البيئة يمكننا أن نبدأ و نبني مشروعنا ، وتشغيل واختبار المشروع ، وحذف الأخطاء في حالة وجودها ، ثم حفظ المشروع للاستخدام في المرة المقبلة، والآن نشرح أجزاء بيئة لغة الفيجوال بيسك:

1-9-1 النافذة الرئيسية : Main Window

ابدأ تشغيل لغة الفيجوال بيسك بالطريقة التي تعلمتها سابقا ، و نلاحظ وجود عدة نوافذ ، والنافذة الرئيسية تستخدم للتحكم في معظم جوانب مشروع الفيجوال بيسك سواء في التصميم أو في التشغيل .

والنافذة الرئيسية تتكون من:

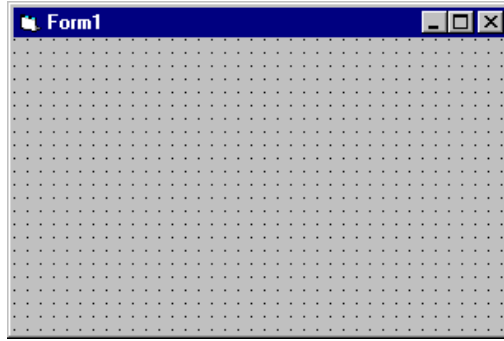
- شريط العنوان

- وشريط القائمة
- وشريط الأدوات

وشريط العنوان يشير لاسم المشروع وطور الفيچوال بيسك الفعّال (التصميم - التعليق - التشغيل) ، أما شريط القائمة فيحوي قوائم منسدلة من خلالها يمكننا التحكم في عمليات بيئة الفيچوال بيسك ، وشريط الأدوات يحوي أزار تزودنا بوصول مختصر لبعض وظائف القوائم ، ننظر الى الأزرار التي استخدمناها لفتح وتعليق و غلق مشروع الفيچوال بيسك.

2-9-1 نافذة المشروع : Form Window

الواقع أن نافذة المشروع من أساسيات بناء مشروع الفيچوال بيسك ، والشكل التالي يوضح نافذة المشروع التي سوف تشاهدها :



وإذا لم نكن نرى نافذة المشروع على الشاشة فيمكننا أن تجري ما يلي لرؤيتها :
ننقر القائمة **View** من القائمة الرئيسية ، ثم ننقر الخيار **Object** أو نلمس المفاتيح **F7+<shift>** في نفس الوقت.

3-9-1 نافذة الأدوات : Toolbox Window

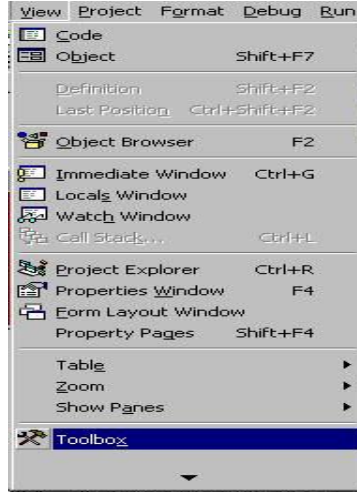
ادوات التحكم هي كائنات معدة لوظائف خاصة ولها مجموعة خصائص ووظائف وأحداث تمامًا كالنوافذ، يأتي Visual Basic6 بفئة أساسية من أدوات التحكم وتجتمع هذه الأدوات داخل مربع واحد يسمى مربع الأدوات وتظل أمامنا في فترة تصميم البرامج تشترك معظم الأدوات في مجموعة من الخصائص بينما تتميز كل أداة بمجموعة من الخصائص التي ترشحها للقيام بوظيفة معينة داخل الواجهة حيث انها الانسب في

الاستعمال. الواقع أن نافذة الأدوات عبارة عن نافذة تحوي أزار تحوي الأدوات التي سوف تستخدمها في مشروعنا ، والجدير بالذكر أن الأدوات يشار إليها بكلمات مثل أهداف objects ، أدوات أو وسائل tools ،

وأدوات تحكم. controls.

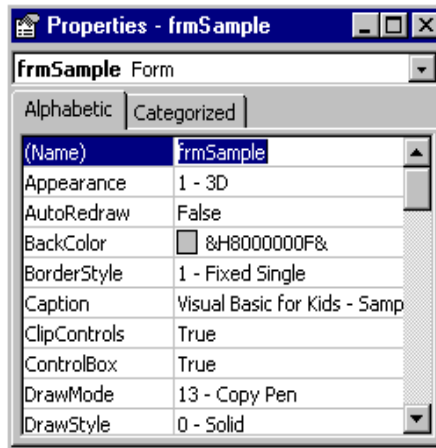


إذا لم تكن نرى نافذة الأدوات على الشاشة ننقر قائمة **View** من القائمة الرئيسية ونختار **Toolbox** ، ننظر الى نافذة الأدوات ونحاول تحديد الأدوات التي المستخدمة في المشروع الذي ناقشناه في المشروع . **Sample**



4-9-1 نافذة الخواص : Properties Window

نافذة الخواص تستخدم في تحديد قيم ابتدائية للأدوات التي نختارها لمشروعنا ، وفي نافذة الخواص نجد في أعلاها قائمة منسدلة تحوي نوع واسم كل الأدوات التي استخدمناها لمشروعنا ، حمل البرنامج sample وننقر تلك القائمة :



نلاحظ أن وجود القائمة المنسدلة في أعلى نافذة الخواص تسمى بقائمة المتحكمات أو قائمة أدوات المشروع ، وهي تظهر اسم كل أداة - حسب تسميتها لها بالمشروع - ونوعها (نص - عنوان - شريط تمرير) ويمكننا أن نستعرض كل أدوات التحكم من خلال شريط التمرير الرأسي لقائمة أدوات التحكم . وتحت قائمة أدوات تحكم مشروعنا نجد قائمة الخواص ، ويمكننا أن تنتقل عبر الخواص بواسطة شريط التمرير الرأسي ، فنجد لليساار اسم الخاصية ، والى يمينها القيمة الابتدائية للخاصية ، ويتضح كثرة

الخواص حيث توجد كثير من الخواص مشتركة بين الأدوات، والقيم الابتدائية توضح أشياء هامة مثل العنوان والاسم والموضع والحجم واللون ... الخ .

النقر على السهم المتجه لأسفل في قائمة أدوات التحكم المختارة وهو موجود أعلى نافذة الخواص يفيد في تحديد كل الأدوات الموجودة على نافذة المشروع form ، و نلاحظ أن من السهل التعرف على الأداة من اسمها يمكننا اختيار أداة تحكم واستعرض خواص تلك الأداة ، وننتقل لأداة أخرى ، ونستمر حتى ننتهي من كل أدوات التحكم في مشروع البرنامج.

5-9-1 نافذة الشفرة Code Window

والآن لننظر لنافذة جديدة وهي نافذة التفسير(البرمجة) ، ونذكر أن الفيچوال بيسك لغة موجهة بالأحداث ، فحينما يكتشف الكمبيوتر حدوث حدث event ، فإن الكمبيوتر يقوم بتنفيذ الإجراء المخصص للحدث **event process** ، وإجراء الحدث يخبر الكمبيوتر الطريقة التي سوف يستجيب بها الكمبيوتر للحدث، والإجراء يحوى جمل الفيچوال بيسك التقليدية ، ونحن سوف نرى كثير من إجراءات الأحداث تلك النافذة

وتوجد طرق عديدة لعرض نافذة الشفرة أو جعلها مرئية :

(a) ننقر أي أداة على نافذة المشروع نقرأ مزدوجاً .

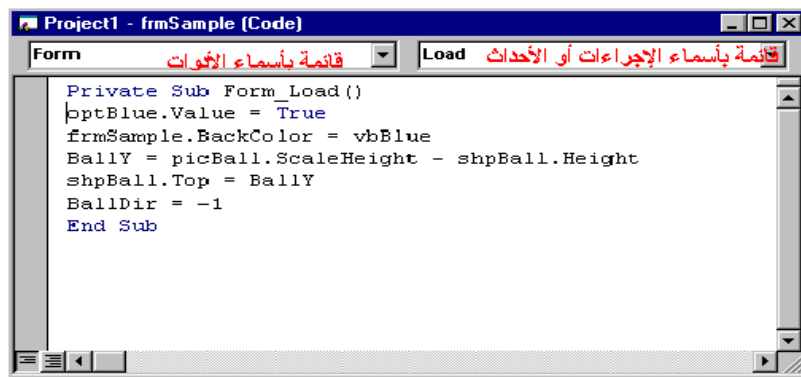
(b) ننقر قائمة **View** من القائمة الرئيسية ونختار **Code** .

(c) نلمس المفتاح **F7** .

(d) ننقر أيقونة الشفرة من نافذة أجزاء المشروع .

بالطرق السابقة ، و نختار أسهلها بالنسبة لنا **Sample** نحاول الوصول لشفرة (برمجة) البرنامج

في نافذة المشروع نجد قائمتين منسدلتين كما بالشكل التالي:



```
Project1 - frmSample (Code)
Form قائمة بأسماء الأدوات Load قائمة بأسماء الإجراءات أو الأحداث
Private Sub Form_Load()
|optBlue.Value = True
|frmSample.BackColor = vbBlue
|BallY = picBall.ScaleHeight - shpBall.Height
|shpBall.Top = BallY
|BallDir = -1
|End Sub
```

وفي أعلى نافذة الشفرة نجد قائمتين منسدلتين الأولى قائمة أسماء الأدوات ، والثانية قائمة بأسماء الإجراءات ، وقائمة أسماء الأدوات شبيهة بقائمة الأدوات الموجودة بنافذة الخواص ، فهي تعرض كل الأدوات الموجودة على نافذة المشروع ، و عندما نختار أحد الإجراءات ، فإن قائمة الإجراءات تعرض كل إجراءات الأحداث الممكنة لتلك الأداة ، وكما نرى بالشكل فلدى اختيارنا الأداة نافذة المشروع Form ، يظهر في القائمة اليمنى الحدث **Load** ، وحدث التحميل **load** يحدث عند تحميل المشروع لأول مرة، وتحت كليهما نجد شفرة البيسك المصاحبة لإجراء الحدث.

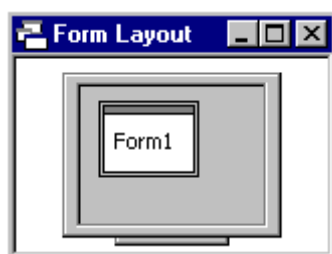
10-1 بناء البرنامج الأول

توجد ثلاث طرق أولية لبناء المشروع :

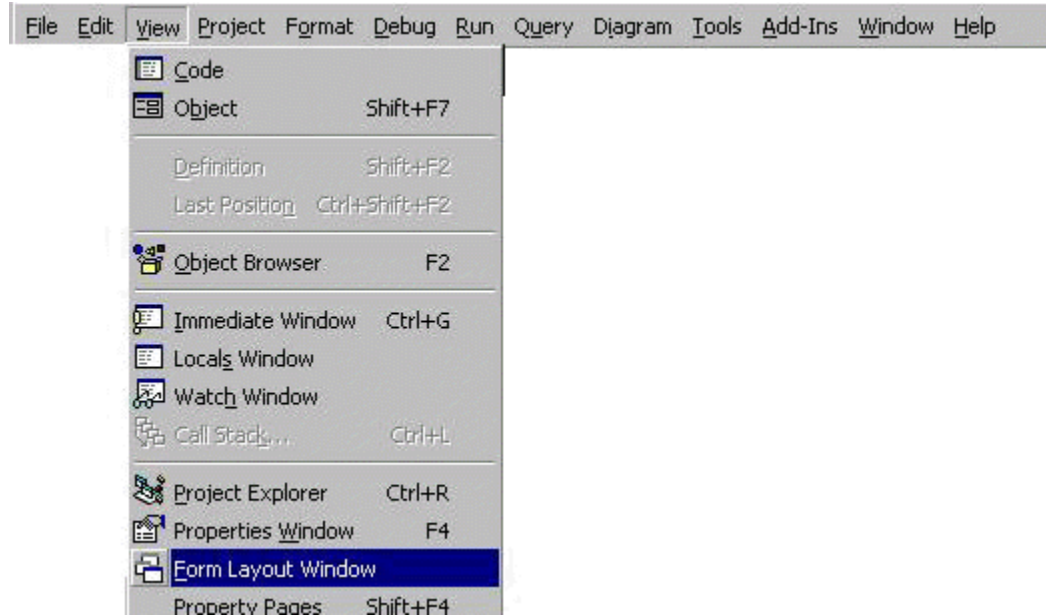
- 1- نضع أدوات التحكم **controls** على نافذة المشروع form .
- 2- نضبط خواص **properties** أدوات التحكم التي نختارها **controls**.
- 3- نكتب إجراءات الأحداث للأدوات **controls** .

وتلك الخطوات تتم في طور التصميم **design** في الفيجوال بيسك ، ونذكر أن طور الفيجوال بيسك يظهر في شريط العنوان في نافذة فيجوال بيسك الرئيسية .

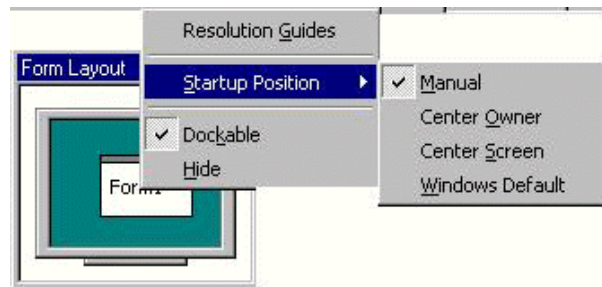
عند البدء بتنفيذ فيجوال بيسك ، و نتعرف على نافذة المشروع . في الإصدار السادس من فيجوال بيسك لا يمكننا أن نحرك نافذة المشروع، فهي مثبتة وفي هذه الحالة يمكننا التحكم فيها من خلال نافذة المخطط **Form Layout Window**:



نضع الماوس على أي جزء ونضغط ونسحب و ننقر زر التشغيل و سنلاحظ الفرق ، فسوف نجد أنه يتحرك معنا، وعند التشغيل نجد الإطار قد وضع في الموضع الذي وضعناه فيه، وإذا كان شكل نافذة المشروع غير موجود بنافذة العرض الجانبية Layout Window موجود ، فننقر القائمة View في القائمة الرئيسية:



كذلك يمكننا أن تضبط موضع الإطار من خلال نقر نافذة العرض الجاني فتنبثق قائمة نختار الخيار Startup Position وهو الموضع عند بدء التشغيل ، فينتج لدينا ثلاث خيارات هم :



□ لتغيير حجم نافذة المشروع نلاحظ أن بها معالج الحجم عند حافة كل ركن ، وعند تمرير المؤشر عبر ذلك المعالج ، حيث يظهر السهم المزدوج ، وهنا يمكننا أن ننقر ونسحب تلك الحافة أو الركن و تضعها في الموضع المطلوب ، نحاول أن نتدرب تحريك وتغيير حجم نافذة المشروع بضع مرات ، فتلك المهارة نحتاجها أيضاً لو نضع الأدوات على نافذة المشروع .

1-10-1 وضع أدوات التحكم على نافذة المشروع

أن أول خطوة في بناء مشروع فيجوال بيسك هي وضع الأدوات التي نحتاجها على نافذة المشروع وفي المواضع المناسبة ، ومن ثم نحدد أدوات التحكم المطلوبة لمشروعنا .

وأدوات التحكم يتم اختيارها من نافذة الأدوات ، وتوجد طريقتين لوضع أدوات التحكم على نافذة المشروع :

1- ننقر الأداة التي نريد وضعها على نافذة المشروع من نافذة الأدوات نقرة مزدوجاً ، فهذا سوف

ينشئ تلك الأداة على نافذة المشروع ولكن بحكم محدد وفي منتصف نافذة المشروع.

2- ننقر الأداة التي نريد وضعها من نافذة الأدوات على نافذة المشروع نقرة واحدة ، ونحرك مؤشر

الماوس عند الموضع المطلوب ونضغط الزر الأيسر مع السحب حتي الحجم المناسب لنا ، و

نلاحظ أن مؤشر الماوس يتغير من السهم إلى السهم المزدوج المتقاطع على شكل حرف + .

وبعد وضع الأداة يمكننا أن نغير موضعها وحجمها أيضاً ، فيمكننا تحريكها بنقر الأداة بالزر الأيسر

للماوس ، حينئذ يظهر لنا معالج الحجم ، حينئذ نسحب للموضع الجديد ، ونحرر مؤشر الماوس بنقر نافذة

الإطار ، ولتغيير حجم الأداة ننقر الأداة بالزر الأيسر ، وعند حافة الأداة سوف يظهر السهم المزدوج نسحب

تلك الحافة إلى الحجم الذي نريده أو نقلص ذلك الحجم بالسحب للداخل ونكرر ذلك مع الحواف الأخرى عند

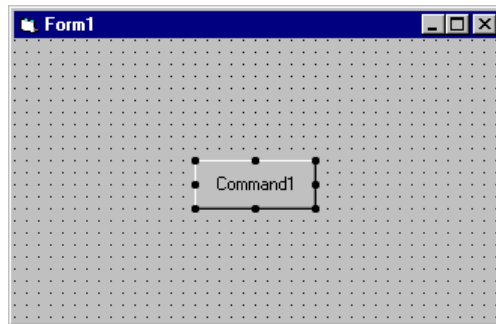
منتصف تلك الحافة .

مثال :

نتأكد من أن فيجوال بيسك نشطة وأننا نرى نافذة المشروع ونافذة الأدوات **Toolbox** ، ومن نافذة الأدوات

ننقر زر الأوامر **command** وننقر أيقونة أداة زر الأوامر نقراً مزدوجاً ، وسوف نشاهد أن الأداة

المصاحبة قد ظهرت على نافذة المشروع :



نلاحظ أن معالج الحجم قد ظهر في منتصفات أضلاع الزر وفي الأركان ، وهذا يعني أن الأداة نشطة **active** ، وإذا نقرت نافذة المشروع فسوف يختفي معالج الحجم ، ونقر الأداة ينشط معالج الحجم ، نحاول أن نحرك الزر من موضعه مع تغيير حجمه عدة مرات .

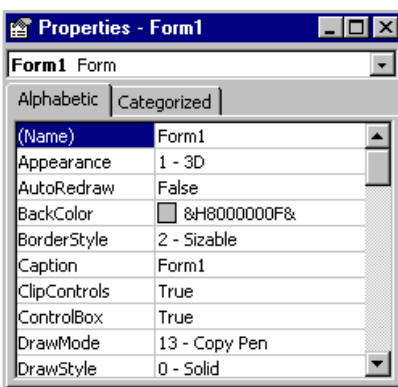
و الآن دعنا نضع زر أوامر آخر باستخدام الطريقة الثانية ، نقر أيقونة الأداة نقرة واحدة ونحرك الماوس للموضع الذي نريد ونضغط الزر الأيسر مع السحب بدون أن رفع الاصبع عن زر الماوس ونستمر الى الموضع الذي نريده ، فالنقرة الأولى تحدد أحد أركان الزر وتحرير الزر يعني تحديد الركن المقابل ويجب أن نعرف تلك الطريقتين ونتمرس عليهما.

ونكرر ذلك مع الأدوات الأخرى مثل أداة العنوان وأداة النص وزر الاختيار ومربع الاختيار أو التحقق . ونحاول أن تجعل أدوات مرتبة وذات مظهر جيد على نافذة المشروع ، فتلك المهارات هامة لتحسين واجهة مشروعنا . و لمعرفة كيفية حذف الأداة من على نافذة المشروع ، ننشط الأداة بنقرها ، وحينئذ سوف يظهر معالج الحجم ، وفي تلك اللحظة نضغط الزر **Del** في لوحة المفاتيح ، وستحذف الأداة النشطة، ويمكننا حذفها أيضاً بلمس المفاتيح **Ctrl + X** معا ولكن الطريقة الأولى أبسط .

1-10-2 ضبط خواص لأداة في طور التصميم :

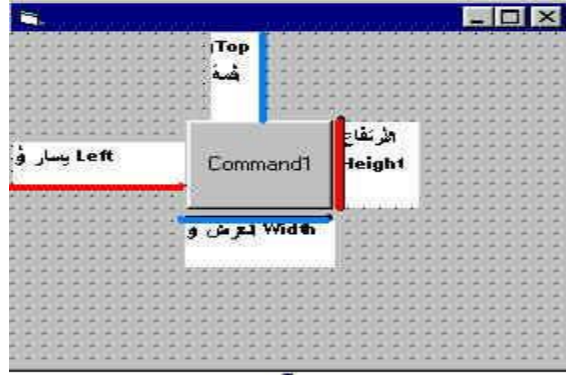
حينما نفرغ من وضع أدوات التحكم على نافذة المشروع ، يلزمنا أن نضبط خواص تلك الأدوات ، ونذكر أن مجرد وضع الأداة على نافذة بحجم وموضع مناسبين يغير من خاصتي الموضع والارتفاع والعرض لتلك الأداة ، فعندما يتم وضع أداة على نافذة المشروع نضع لها فيجوال ببسك خواص ابتدائية تسمى بالقيم الفعالة **default** مثل خواصها الهندسية التي تحكم الحجم والموضع ، ولكن القيم الابتدائية قد لا تناسبنا ، ونريد تغييرها ، ويمكننا أن نفعل ذلك من خلال نافذة الخواص **Properties Window** .

نشغل فيجوال ببسك ، وننقر قائمة **ملف File** ، وننقر الخيار مشروع جديد **New Project** ، وسوف نُسأل عن نوع المشروع ، فنختار **Standard EXE** ، وهنا يظهر لدينا نافذة مشروع خاوية على الشاشة ، نبحث عن نافذة الخواص **Properties Window** ، فإذا لم نكن نراها فنلمس مفتاح الوظائف **<F4>** . وننقر زر ترتيب الخواص هجائياً **Alphabetic** إذا كان زر تصنيف الخواص **Categorized** هو النشط ويصبح شكل نافذة الخواص كما يلي :

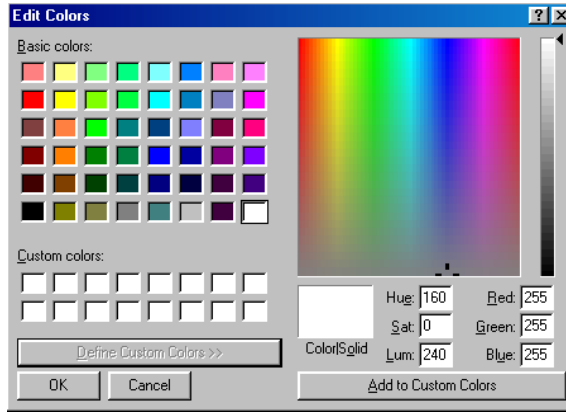


ونذكر أن المربع أو الصندوق الذي يعطى نافذة الخواص يسمى قائمة الأدوات المختارة **control list** يخبرنا بالأدوات الموجودة على نافذة المشروع ، وكما نرى الآن فالقائمة تحوي أداة واحدة هي أداة نافذة المشروع لأننا لم نضع أي أداة أخرى بعد على نافذة المشروع ، فنحاول أن نتعرف على خواص نافذة المشروع . ولتكبير حجم نافذة المشروع لدينا خواص أربع تحدد حجم المستطيل ، ننزل لأسفل نافذة الخواص باستخدام شريط التمرير وابحث عن خاصية الارتفاع **Height** ، وهي تعبر عن ارتفاع نافذة المشروع بوحدات تسمى **twips** فكل 1440 تويب تساوي بوصة طولية وبالتالي نقسم الارتفاع على 1440 لتحصل على ارتفاع نافذة المشروع بالبوصة ، وبالمثل خاصية العرض **Width** مقاسة بالتويب . نحاول تغيير حجم - مساحة- نافذة المشروع و نشاهد التغيرات التي تحدث على خاصيتي الطول والعرض ، أما الخاصية يسار **Left** فتحدد بُعد نافذة المشروع عن حافة الشاشة اليسرى ، أما الخاصية قمة **Top** فتحدد بُعد نافذة المشروع عن حافة الشاشة العليا ، وباستخدام نافذة المخطط **Layout Window** حرك نافذة المشروع وشاهد التغيرات التي تحدث على الخاصيتين يسار **Left** وقمة **Top** . أو ننقر على أي من تلك الخاصيتين في نافذة الخواص وغير قيمتهما و نلاحظ ما يحدث لنافذة المشروع . ومن ثم فالخواص الأربع يسار **Left** و قمة **Top** وعرض **Width** وارتفاع **Height** يصفون الموضع والحجم .

نمرر شريط التمرير للخاصية لون الخلفية **BackColor** ، أن تلك الخاصية تحدد لون خلفية نافذة المشروع - الأداة - ونلاحظ أن قيمة تلك الخاصية هي **&H000000F&** وهذا في الواقع هو رقم اللون الرمادي



ونلاحظ شيئين فلدی نقرنا للزر نظام System وهنا نجد شريط أزرق على وجه الزر ونجد لونه رمادي ، ويمكننا أن نختار لون خلفية نافذة المشروع من عدة ألوان ، والشئ الثاني هو وجود رسالة أسفل نافذة الخواص تحدد وظيفة الخاصية ، والآن ننقر زر الألوان Palette و سوف تظهر لدينا نافذة الألوان التالية - ننقر أحد تلك الألوان ون نلاحظ لون نافذة المشروع .



هناك خصائص تتحكم في الوان الكتابة وهي:

ForeColor: لتحديد لون النص في الخلية العادية.

ForeColorFixed: لتحديد لون النص في الخلية الثابتة.

ForeColorSel: لتحديد لون النص في الخلية المعلمة.

هناك مجموعة خاصة بتحديد لون الخلفية في مناطق مختلفة من الجدول:

BackColor: للخلية العادية.

BackColorFixed: للخلية الثابتة.

BackColorSel: للخلية المعلمة.

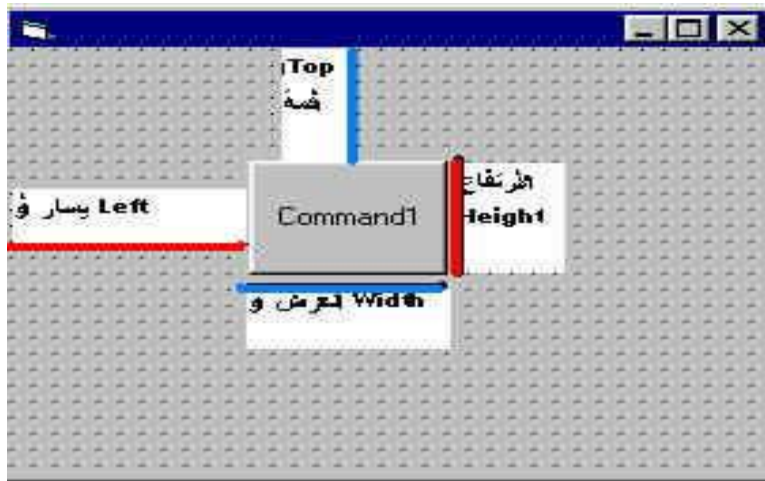
BackColorBkg: للمنطقة الخلفية من الجدول التي لا تحتوي على أية خلية

نمرر الشريط لخاصية العنوان **Caption** ، وهي الخاصية التي تحدد العنوان الذي سوف يظهر في شريط عنوان نافذة المشروع ننقر تلك الخاصية ونكتب أي شئ و نلاحظ التغير الذي يحدث على شريط عنوان نافذة المشروع .

مثال :

نبدأ مشروع فيجوال بيسك ونغير موضع وحجم نافذة المشروع ، ونتحقق من بعدها عن حافة الشاشة اليسرى **Left** والحافة العليا **Top** ، وعرضها **Width** وارتفاعها **Height** ونضبط خاصية لون الخلفية **BackColor** ، وخاصية العنوان **Caption** .

نضع أداة زر الأوامر **command button** على نافذة المشروع ونضبط حجم وموضع الزر .
ننقر الزر أو اختاره قائمة الأدوات الموجودة على نافذة المشروع ، وهو أيضاً مستطيل الشكل ، وسنجد أن العرض **Width** وارتفاع **Height** مقاستين بالتويب وكذلك اليسار **Left** والقيمة **Top** ولكن نلاحظ أن الخاصية يسار **Left** تحدد بعد حافة زر الأوامر عن حافة نافذة المشروع اليسرى وليس الشاشة وكذلك الخاصية قمة **Top** تحدد بعد زر الأوامر عن حافة نافذة المشروع العليا ، فالقياس هنا بالنسبة لنافذة المشروع وليس الشاشة وكل من تلك الوحدات أمامها أرقام بالتويب ما لم تكن غيرنا نمط القياس من الخاصية **Scale Mode** ، وبالنسبة لزر أوامر واحد تكون خواص زر الأوامر هي :



1-10-3 تسمية أدوات التحكم :

أن أكثر الخواص أهمية هو اسمها Name وهي تختلف عن خاصية العنوان Caption ، فاسم الأداة هو الاسم الذي سوف يتم نداء الوظائف المخصصة للأداة من خلاله ، ويحسن أن يحوي اسم الأداة شقين الأول هو نوعها والثاني هو وظيفتها ، فقد يكون لدينا عدد كبير من أزرار الأوامر مثل مشروع الآلة الحاسبة ولكن لكل زر وظيفة محددة مثل القسمة والجمع الخ

وقد اقترحت شركة ميكروسوفت اختصارات للأدوات وهي :

الاختصار	الأداة	
frm	Form	نافذة المشروع
cmd	Command Button	زر الأوامر
lbl	Label	العنوان
txt	Text Box	خانه النص
chk	Check B	خانه التحقق
opt	Option Button	زر الاختيار

وبعد اختيار اسم الأداة نختار الجزء الأول كما بالاختصارات السابقة - وهي اختيارية - مبدوءة بحرف كبير ، ونختار الوظيفة أيضا مبدوءة بحرف كبير ، ويجب أن لا يزيد اسم الأداة عن 40 حرف شريطة أن لا يكون من بينها أحد أحرف العمليات ، ولكن نحاول أن نجعل الأسماء قصيرة قدر الإمكان فهذا يوفر وقت كتابة الأسماء الطويلة .

1-11 ضبط الخواص في طور التشغيل :

ولتوضيح أهمية اختيار أسماء فعلية لأدوات التحكم ، دعنا ننظر الى إحدى مهام الفيچوال بيسك ، كذلك يمكننا أن تغير الخاصية أثناء التشغيل فمثلاً في المشروع **Sample** يمكننا تغيير لون خلفية نافذة المشروع بنقر أحد أزرار الاختيار ، فهو يغير خاصية لون الخلفية **BackColor** بالون المخصص في وظيفة الزر، ويمكن تغييرها من خلال جملة الفيچوال بيسك :

قيمة الخاصية الجديدة = اسم الأداة . اسم الخاصية

ControlName.PropertyName = PropertyValue

وهي تبدأ باسم الأداة ونقطة (نقطة العلامة العشرية) واسم الخاصية التي نرغب في تغييرها وتلك الخاصية يجب أن تكون من ضمن خواص الأداة الموجودة بنافذة الأدوات و علامة تساوي ثم القيمة الجديدة للخاصية وشفرة تغيير نافذة المشروع **Sample** للون الأزرق هي :

لون فيجوال بيسك الأزرق = لون الخلفية . نافذة المشروع

frmSample.BackColor = vbBlue

نلاحظ أن الأسماء الفعلية تجعل الشفرة مفهومة كما نرى، frm sample تعني نافذة المشروع ، Backcolor تعني لون الخلفية ، vb اختصار لفيجوال بيسك visual basic ، و Blue تعني معا لون الفيجوال بيسك الأزرق .

12-1 كيفية استخدام أسماء الأدوات في إجراءات الأحداث :

وثمة ميزة أخرى لاختيار أسماء فعلية للأدوات وهي وضوحها في إجراءات الأحداث ، والواقع أنه عند وضعنا للأداة على نافذة المشروع ، فإن كل إجراءات الأحداث الممكنة لتلك الأداة يمكننا أن نضيفها لمشروعنا ، وكما رأينا أن كل إجراءات الأحداث يمكن مشاهدتها في نافذة الشفرة أو ما يعرف بمحرر فيجوال بيسك ، وعادة يكون التركيب الهيكلي لتلك الإجراءات كما يلي :

سطر العنوان الرئيسي: **Private Sub ControlName_EventName()**

[شفرة البيسك التي تنفذ وظيفة الإجراء]

سطر الختام: **End Sub**

نلاحظ أن سطر العنوان يضم اسم الأداة ، ومن ثم فأسماء الأدوات الفعلية تمكننا من تحديد كل إجراء حدث بسهولة .

13-1 كتابة إجراء الحدث :

والواقع أن الخطوة الثالثة هي بناء تطبيق فيجوال بيسك لأداة موجودة على نافذة المشروع ، وكتابة إجراء حدث نستخدم نافذة الشفرة - محرر فيجوال بيسك، ننقر الأداة نقرأ مزدوجاً أو نقر أيقونة نافذة الشفرة من نافذة أجزاء البرنامج أو اختيار الخيار Code من قائمة View ، أو لمس المفتاح F7 .
وكما ذكرنا أنفاً فعند وضع الأداة على نافذة المشروع يمكننا دمج أي من أحداث تلك الأداة بمشروعنا من خلال نافذة الشفرة **code window** ، فكل أداة لها العديد من الأحداث ، وبالطبع فأنت لا نحتاج لكل أحداث تلك الأداة إنما نحتاج فقط لما يناسب وظيفة مشروعنا ، وكما تعلمت سابقاً فصيغة الإجراء هي :

سطر عنوان الإجراء `Private Sub ControlName_EventName()`

[شفرة الفيجوال بيسك]

سطر غلق الإجراء `End Sub`

وكلمة 'Private Sub' تعني إجراء خاص ، أو برنامج فرع للأداة عند اقترانها بحدث معين ، والواقع أن شفرة البيسك تعد العمل الابتكاري في المشروع ، ونذكر أن نقص حرف أو علامة تنصيص أو أحد رموز العمليات سوف يسبب أخطاء وبالتالي لن يعمل برنامجنا وبالتحديد الجزء الذي يحوي الخطأ ، وهذا النوع من الأخطاء يسمى بالخطأ التركيبي `syntax error` ، فكتابة البرنامج تتطلب دقة متناهية .
ومن ثم فكتابة إجراءات الحدث :

- نحدد الحدث الذي نرغب في أن يستجيب برنامجنا له .
- نحدد كيفية الاستجابة للحدث .
- نترجم الاستجابة لشفرة فيجوال بيسك .
- نبحث عن إجراء الحدث في نافذة الشفرة .
- نكتب شفرة البيسك .

والمثال التالي يوضح ذلك :

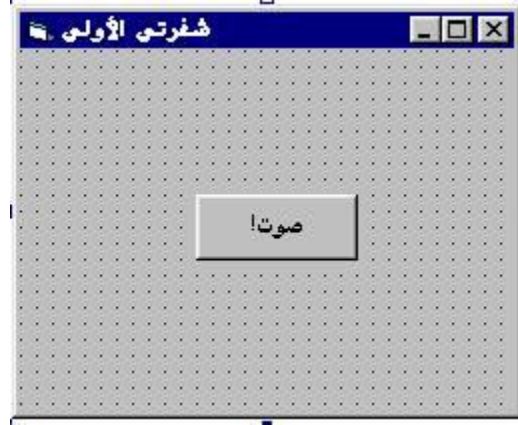
مثال :

- نضع زر أوامر واحد على نافذة المشروع .
- نضبط خاصية اسم نافذة المشروع `Name` للاسم `frmFirstCode` .
- ونضبط عنوان نافذة المشروع من الخاصية على العنوان "شفرتي الأولى"

- ضبط خاصية زر الأوامر **Name** - تحدد اسم الشفرة للزر - على الاسم **cmdBeep** .
- ضبط خاصية زر الأوامر **Caption** - نحدد عنوان الزر أو وظيفته - على

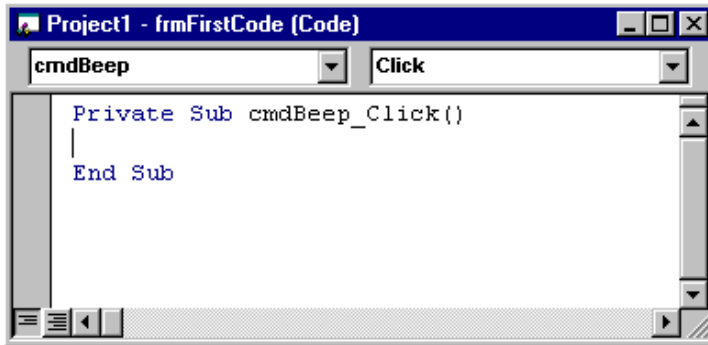
الاسم صوت!

حينئذ يصبح شكل نافذة مشروعنا كما يلي :

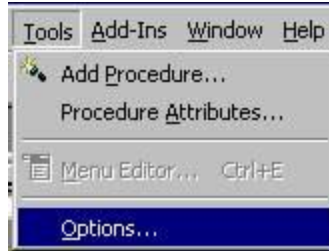


ونحن نرغب في كتابة إجراء حدث واحد يستجيب لحدث نقر **Click** زر الأوامر **command button** فنحن نرغب في أن يصدر الكمبيوتر صوت بيب **beep** من سماعته الداخلية لدي نقر المستخدم على ذلك الزر .

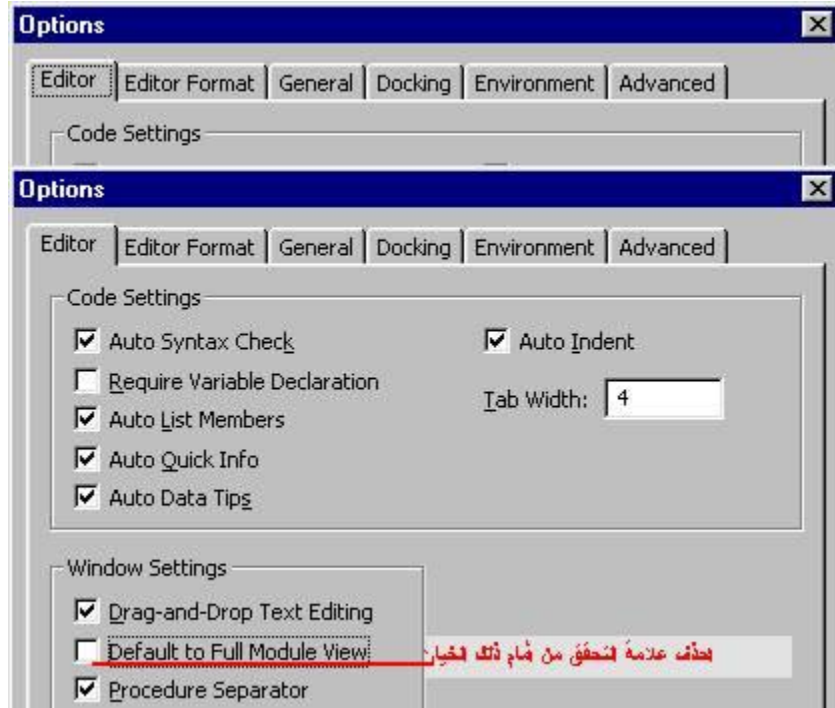
لكي تجعل نافذة الشفرة أو محرر فيجوال بيسك مرئي: F7 نلمس المفتاح



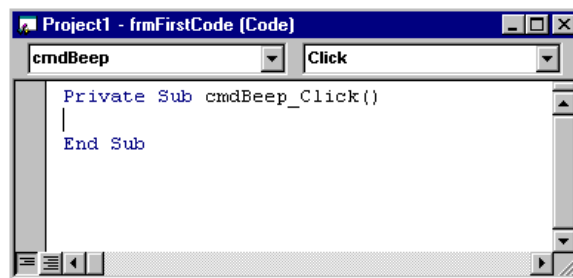
وبما يكون شكل نافذة المحرر أكبر من ذلك ففي بيئة فيجوال بيسك يوجد نمط عرض كامل **full-module** يرينا مكان قائمة أحداث الإجراءات واحد بعد الآخر وليس كما نراها هنا ولكي نحصل على نمط العرض الكامل ننقر القائمة **Tools** ثم نختار الخيار **options** كما بالشكل التالي :



تظهر لنا النافذة التالية :



واحذف علامة الصح من أمام الخيار **Default to Full Module View** أن وجدت .
 وإذا لم نجد اسم زر الأوامر command button في قائمة الأهداف التي تعلقو نافذة الخواص ننقر ونختار
 اسم الزر cmdBeep ، ويصبح شكل نافذة الشفرة كما يلي :



و نلاحظ إجراء حدث النقر click للزر **cmdBeep** ، وسوف نحتاج قائمة الإجراءات كثيراً للوصول للإجراء الذي نريد ، مع ملاحظة أن الإجراءات مرتبة ترتيباً هجائياً ، وبين رأس الإجراء ونهايته نكتب شفرة البيسك التي تنفذ وظيفة إصدار الصوت .ونكتب السطر التالي :

Beep

وتلك التعليمة تخبر الكمبيوتر أن يصدر دقة جرس ، والواقع فقد خطوت الخطوة الأولى في البرمجة بلغة الفيجوال بيسك

```
Private Sub CmdBeep_Click()
```

```
Beep
```

```
End Sub
```

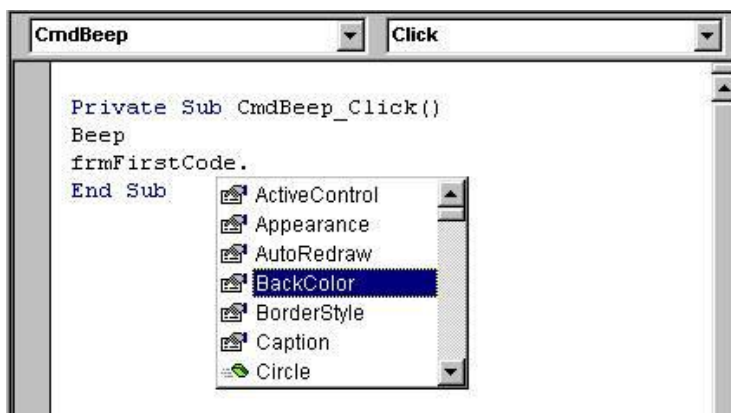
والآن برنامجنا جاهز للتشغيل Run ننقر زر البدء أو نختار start من قائمة Run أو نلمس المفتاح F5 لنرى ناتج برنامج الأول .



ونقر الزر يخبر الكمبيوتر أن يذهب الى إجراء الحدث المحدد لحدث النقر **cmdBeep_Click** ويقوم بتنفيذ شفرته و هي الأمر **Beep** . وربما رغبت في أن يغير الكمبيوتر لون خلفية نافذة المشروع فأضف السطر التالي بعد الأمر **Beep** :

```
frmFirstCode.BackColor = vbBlue
```

والواقع أن فيجوال بيسك لدي كتابتنا لاسم الأداة ووضع نقطة تنبثق قائمة تحوي الأوامر الخواص المتاحة لتلك الأداة كما يلي :



ننقر الزر و نلاحظ ما سوف يحدث لنا فذة المشروع



14-1 حفظ مشروع الفيجوال بيسك :

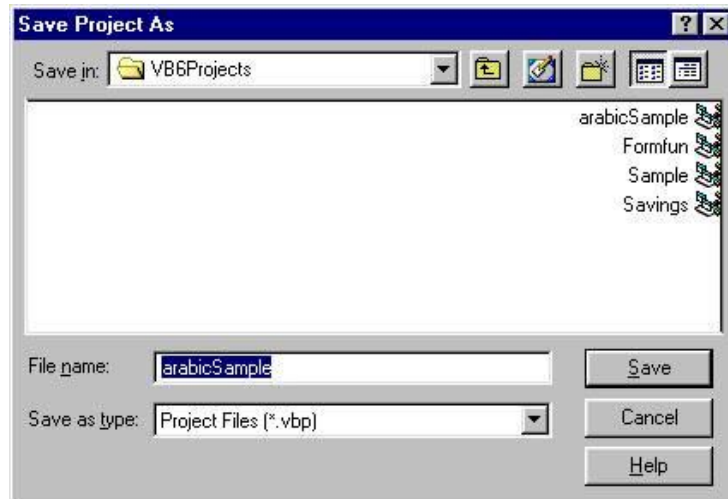
تعلمنا كيفية فتح وتشغيل وغلق مشروعات فيجوال بيسك سبق حفظها أو تخزينها ، ولكننا لم نتحدث عن كيفية حفظ المشروع للاستخدام في المرات القادمة ، والآن لنفرض اننا بدأنا بناء مشاريعنا ونحتاج لحفظها ، والواقع أن العملية سهلة ، وسوف نستخدم نافذة فيجوال بيسك الرئيسية أو شريط الأدوات ، فابحث في شريط الأدوات عن شكل قرص مرن ، فننقر هذه الأيقونة يجعل فيجوال بيسك تحفظ مشروعنا على أحد الأقراص المتاحة ومن ثم يمكننا تشغيل المشروع في المستقبل بل وتطويره وتعديله ، وتلك الأيقونة تسمى بحفظ المشروع **Save Project** ، والأيقونة التي تفعل ذلك تأخذ الشكل التالي :



ولكن ننقر تلك الأيقونة يعطي نتائج مختلفة طبقاً لحالة المشروع ، فالمشروع الذي سبق حفظه على القرص الفعال لا يخرج رسائل بل يحفظ التعديلات مباشرة . فإذا كنا نعمل في مشروع جديد لم نحفظه بعد فسوف يخرج لنا فيجوال ببسك النافذة التالية :



وتلك النافذة تستفسر عما إذا كنا نرغب في حفظ نافذة المشروع بالاسم الذي اقترحتة علينا أو أن نضع اسم آخر لذلك هيئت لنا الأمر وظللت الاسم الذي اقترحتة **Form1** ، ونذكر أن ذلك الملف سوف يحفظ بالامتداد **frm** فنحرك للمجلد الذي نريد حفظ ملفنا به وخصص اسم للملف - يجب أن يكون الاسم يحمل معني ربما يرتبط بوظيفة البرنامج - وننقر الزر **Save** ولكن بعد أن ننتهي من عملية الحفظ تلك سوف تظهر لنا نافذة أخرى ، فالواقع أننا حفظنا جزء من المشروع وسوف يطلب منا فيجوال ببسك حفظ الجزء الثاني لذلك تظهر لنا النافذة التالية :



انواع الأحداث :

في الواقع أن نافذة المشروع تعمل كحاوية للأدوات الأخرى، ولكنها تدعم أحداث أي تتأثر أو تستجيب أيضاً لبعض الأحداث ، فيمكنها أن تستجيب لبعض تفاعلات المستخدم ، وسوف نهتم بحدثين فقط:

الحدث	الوصف
Click	وهو الحدث الذي يتحقق لدي نقر المستخدم لنافذة المشروع .
Load	وهو حدث يحدث لدي تحميل نافذة المشروع لأول مرة لذاكرة الكمبيوتر ، وربما يكون ذلك الوقت مناسب لتخصيص قيم ابتدائية للخواص عامة ولقيم المتغيرات الابتدائية .

سبق أن ذكرنا أن اسم الأداة مهم لأنه يدخل في كل إجراءات الأحداث ، ولكن اسم **Name** نافذة لا يؤثر بنفس الطريقة ، إذ تأخذ كل الأحداث اسم الأداة كما يلي :

Form_EventName

فمثلاً :

```
Private Sub Form_Load()
```

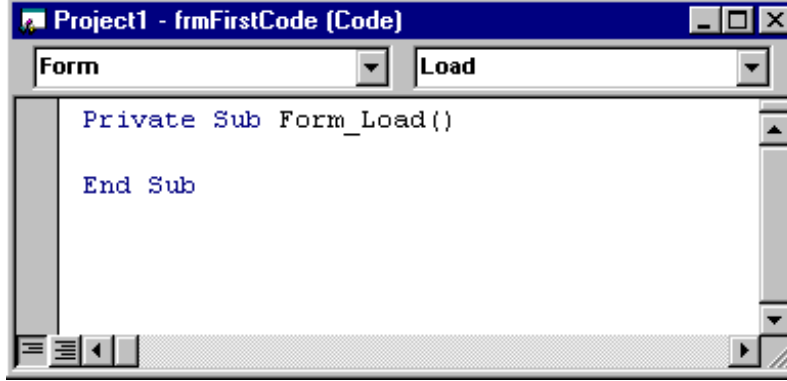
```
End Sub
```

و

```
Private Sub Form_Click()
```

```
End Sub
```

وهذا الاسم سيظل كذلك حتى ولو سميت نافذة المشروع بأي اسم آخر فلو فرضنا أن اسم نافذة الإطار **frmFirstCode** فإن حدث التحميل سيظهر في نافذة الشفرة كما يلي :



15-1 أداة زر الأوامر command button

تعتبر أداة زر الأوامر من أكثر أدوات الفيجوال بيسك استخداماً ، حيث يمكن استخدامها لبدء أو تعليق أو إيقاف ، ... الخ من العمليات ، وشكل زر الأوامر الذي تختاره من نافذة الأدوات يكون على الشكل التالي :

In Form :



On Tool Box:



الخواص :

لنتأمل بعض من خواص زر الأوامر التالية :

الخاصية الوصف

Name : وهو الاسم المستخدم لتحديد وتمييز زر الأوامر ، وعادة يضم أوله الحروف الثلاثة **cmd** .

Caption : العبارة التي تظهر كعنوان لزر الأوامر

Font : وهي الخاصية التي تحدد نوع الخط Arial أو Times New Roman ... الخ ، وحجم الخط

72 - 8

Left : مقدار بعد حافة زر الأوامر اليسرى عن حافة نافذة المشروع اليسرى.

Top : مقدار بعد حافة زر الأوامر العلوية عن حافة نافذة المشروع العلوية بوحدة القياس .

Width : مقدار عرض زر الأوامر بوحدة القياس (تويب - سم ملي - بوصة ...)

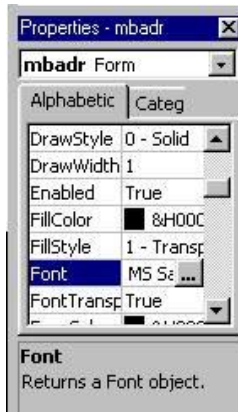
Height : مقدار ارتفاع زر الأوامر بوحدة القياس (تويب - سم ملي - بوصة ...)

Enabled : وهي خاصية تحدد ما إذا كان الزر سوف يستجيب لأحداث المستخدم من عدمه ، وتأخذ قيمتين True أو صحيح وهنا يكون يستجيب الزر ، والقيمة الثانية False وهنا يبهت شكل الزر ولا يستجيب للمستخدم في طور التشغيل.

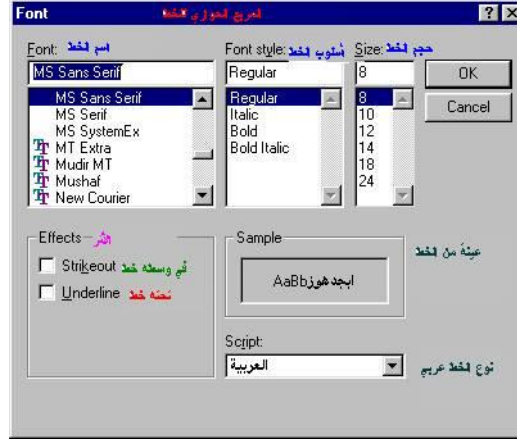
Visible: وهي خاصية تحدد ما إذا كان الزر سوف يكون مرئياً أو منظوراً على الشاشة من عدمه ، وهي تأخذ قيمتين وهما True وهذا يعني أن الزر يشاهده المستخدم ، و False وتعني أن الزر سوف يكون غير مرئي .

مثال :

نشغل فيجوال بيسك ، ونبدأ مشروع جيد ، نضع أداة زر الأوامر على نافذة المشروع ، ونحرك الزر في أماكن متعددة من نافذة المشروع ، و نلاحظ التغييرات التي تحدث على الخاصيتين قمة Top ، يسار Left ، وغير حجم الزر و نلاحظ تغيير قيمتي الخاصيتين عرض Width ، والارتفاع Height ، وغير خاصية العنوان Caption . والخط المكتوب المعبر عن عنوان زر الأوامر يتمتع بخاصية الخط Font -كثير من الأدوات تتأثر بالخاصية خط Font - فالخاصية خط Font يحدد خصائص الخط الذي يظهر كعنوان لزر الأوامر ، فعندما ننقر على الخاصية خط Font في نافذة الخواص يظهر زر يسمى زر انتقال فجائي ellipsis سوف يظهر الى اليمين من النافذة أمام الخاصية Font

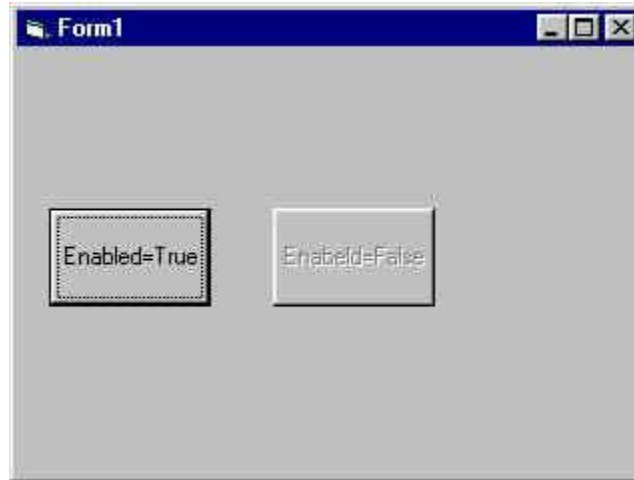


ونقر هذا الزر يؤدي لظهور المربع الحواري للخطوط أو نافذة الخط **Font Window** ومن هذه النافذة يمكننا أن نختار 3 أشياء هم اسم الخط Font ، وأسلوب الخط (مائل - غامق - عادي - غامق مائل) ، وحجم الخط ، ويمكننا اختيار أثر جانبي مثل وضع خط تحت النص أو وضع خط في منتصف كلمات النص .



وتوجد خاصيتين مهمتين لزر الأوامر وهما الخاصية Enabled والخاصية مرئي أو منظور Visible وقيمتها إما صحيح True (نعم) أو خطأ False (لا) والكثير من الأدوات الأخرى تتمتع بتلك الخاصية .

فإذا كانت قيمة الخاصية Enabled هي True فإن المستخدم لن يكون قادراً على التعامل مع الزر علامة ذلك هو ظهور الزر بلون باهت ، ولن يستجيب للمستخدم



وإذا كانت قيمة الخاصية Enabled هي False فلن يستطيع المستخدم التعامل مع الأداة ، وتصبح الأداة باهته .

أما الخاصية مرئي أو منظور Visible فلا تعرف إلا شيئين ، وهما إما منظور Visible = True وفي هذه الحالة يكون الزر مرئياً ، أو منظور Visible = False وفي هذه الحالة يختفي تماما الزر من على نافذة المشروع .

16-1 أداة العنوان Label Control :

تقوم أداة العنوان بعرض معلومات لا يمكن للمستخدم أن يصححها أو يحدفها بشكل مباشر ، وتستخدم غالباً في عرض عناوين أو أسماء الأدوات الأخرى ، أو تستخدم في عرض نتائج عمليات الكمبيوتر ، وأداة العنوان نختارها من نافذة الأدوات.

الخواص :

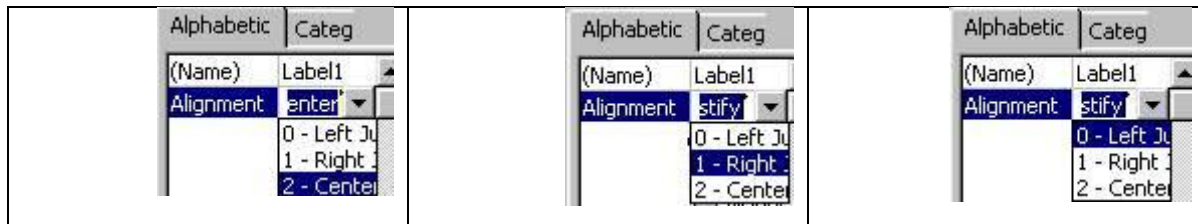
تتمتع أداة العنوان بكثير من الخواص لعل أهمها :

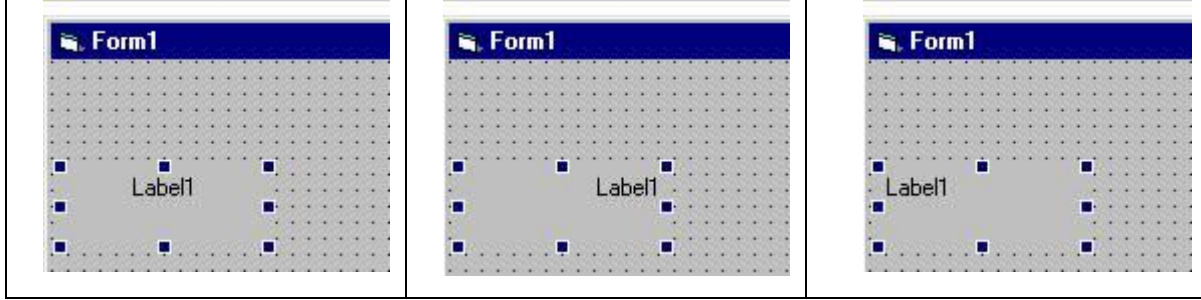
اسم الخاصية	الوصف
Name	اسم يستخدم لتحديد أداة العنوان ، وعادة يبدأ بالحروف الثلاث lbl .
Caption	العبرة التي سوف تظهر في أداة العنوان.
Font	يضبظ أسلوب الخط ونوعه وحجمه الذي سوف يظهر في العنوان.
Alignment	تحدد يظهر النص من اليسار أم من اليمين أم من المنتصف ، وهذا مهم في اللغة العربية
BackColor	يضبظ لون خلفية أداة العنوان .
ForeColor	يضبظ لون الخط الذي سوف يظهر في أداة العنوان.
BorderStyle	يحدد نوع الحافة هل هي Fixed أي مجسمة أم لا none
Left	يحدد المسافة التي تبعتها حافة أداة العنوان اليسرى عن حافة نافذة المشروع اليسرى .
Top	المسافة التي تبعتها حافة الأداة العلما عن حافة نافذة المشروع العلما .
Width	عرض أداة العنوان بوحدة القياس التي نختارها من خواص نافذة المشروع وهي Scale Mode والقيمة الفعالة لها هي التويب

ارتفاع أداة العنوان بوحدة القياس النشطة.	Height
يحدد هل تظهر الأداة أم لا	Visible
وهي تتيح تكيف أداة النافذة طبقاً لعدد حروف عبارة العنوان ولها قيمتان True وتؤدي للتكيف الأوتوماتيكي مع حجم عبارة العنوان ، والقيمة False تجعل حجم الأداة ثابت على نافذة الإطار .	AutoSize
ولها قيمتان True وتؤدي لبهتان الأداة و تعني عدم تمكين ، بينما False تؤدي الى ظهور الأداة بشكله الطبيعي وتمكين المستخدم من التعامل معها بالأحداث .	Enabled
وهي خاصية تقبل حرفية ، وهي تحدد العبارة التي سوف تظهر على الشاشة كاستجابة لتمرير مؤشر الماوس على الأداة .	TopTipText

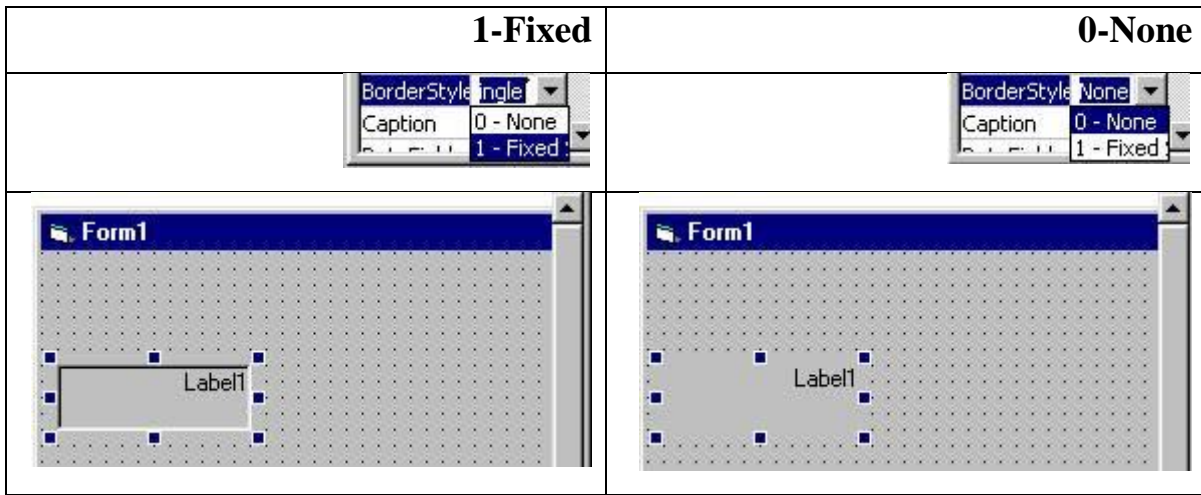
مثال :

نضع أداة عنوان Label على نافذة المشروع ، نغير حجمها وموضعها للحجم والموضع الذي نريد، ونضبط خاصية العنوان ، ونجرب خطوط مختلفة ، و خاصية التنسيق أو المحاذاة alignment - و نلاحظ الفرق بين الخيارات الثلاثة لهذه الخاصية يمين Right - 1، ووسط Center - 2، ويسار Left - 0 (نلاحظ عدم وجود اتجاه عمودي للنص ، فشريط العنوان يظهر أفقياً فقط .





و نلاحظ الفرق بين احتمالي `BorderStyle` ، فلدَى ها القيمة (0-None) والقيمة (1-Fixed) فالأولي تجعل أداة العنوان مسطحة أما القيمة 1 فتجعل الحافة مجوفة كما يلي :



غير كل من خواص الخاصيتين لون الخلفية `BackColor` ، وخاصية `ForeColor` ، ويجب أن نختار ألوان للمقدمة (لون الخط) والخلفية مناسبة حتى لا تنفر العين بل يحسن اختيار ألوان جذابة للعبارة التي تكتبها في خاصية العنوان `Caption` .

فأكثر خاصية مستخدمة للأداة `Label` هي خاصية العنوان `Caption` ، فكل العبارة التي سوف تظهر في أداة العنوان يتم وضعها في طور التصميم أو أثناء البرمجة من خلال تلك الخاصية ويمكننا استخدام شفرة البيسك التالية لوضع تلك العبارة كعنوان لأداة العنوان :

`lblExample.Caption = "هذا عنوان أداة العنوان"`


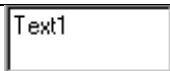
الأحداث :

لأداة العنوان Label حدث هام وهو حدث النقر (لها أحداث أخرى) :

الحدث	الوصف
Click	هذا الحدث يحدث عندما ينقر المستخدم على أداة العنوان بالماوس .

17-1 أداة النص Text Box Control

أداة النص تستخدم لعرض المعلومات المدخلة في طور التصميم أو طور التشغيل ، أو التخصيص بدون إجراء حدث . وأداة النص **text box** شكلها على نافذة الأدوات Toolbox يختلف عن شكلها على نافذة المشروع كما يوضحها الشكل التالي :

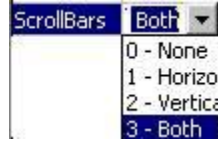
In Toolbox:	On Form (default properties):
	

ولكن نذكر أن أداة النص تشبه أداة العنوان ، ولكن ان أداة العنوان Label هي أداة اخراج وعرض محتوى فقط بينما أداة النص TextBox تستخدم للاخراج المحتوى وايضا كنوع من ادوات ادخال قيم للبرنامج (يعني ادخال واخراج المحتوى) وايضا أداة النص يمكن للمستخدم أن يغيرها بينما أداة العنوان لا يمكنه أن يغير المعروض في أداة العنوان مباشرة ، والواقع أنه يمكن وضع الأدوات على الشاشة بحيث لا يستطيع المستخدم التفريق بينهما إلا من تلك الناحية فيمكننا تاشير أي جزء من أداة النص بينما لا يمكننا فعل ذلك في أداة العنوان كما يلي :



فكما نرى أمكن تاشير كلمة النص في أداة النص ، بينما لن يمكننا فعل ذلك في أداة العنوان ، وثمة اختلافات أخرى لعل أبرزها أن أداة النص يمكننا أن نختار لها خاصية السطور المتعددة Multiline وتأخذ

قيمتين True وهي تتيح تعدد السطور ، والقيمة False تعرض كل النص في سطر واحد ، ومن ثم يمكننا أن نختار نوع شريط التمرير من الخاصية Scrolbars :



وتأخذ اربع قيم o-None أي بدون أشرطة تمرير ، 1-Horizontal وهذا يتيح شريط تمرير أفقي ، 2-Vertical وهذا يتيح شريط تمرير رأسي ، بينما 3- Both يتيح شريطين أحدهما أفقي والثاني رأسي في نفس الوقت .



وشريط التمرير غير متاح لأداة العنوان ، وثمة فارق آخر هو في حجم المعلومات فأداة النص يمكنك من استيعاب عدد معين من الحروف بحد أقصى 64 ك. ب. ، وعموماً فأبرز خواصها ما يلي :

الخواص Properties

الخاصية	الوصف
Name	اسم يستخدم لتحديد أداة النص ، واقتراح أن يبدأ بالحروف الثلاث txt
Text	العبرة التي سوف تظهر في أداة النص ، والتي يمكن للمستخدم تعديلها ، وهي تشبه خاصية العنوان في أداة العنوان Label
Font	تضبط اسم الخط ونوعه وحجمه .
Alignment	وهي تحد ما إذا كان النص سوف يحازي أو ينسق من اليسار left أو من اليمين right أو يو نضع في المنتصف Center .
MultiLine	خاصية السطور المتعددة تحدد ما إذا كان النص سوف يظهر في سطور متعددة أم في سطر واحد ولها قيمتين True وتعني ظهور النص في سطور متعددة ، والقيمة False تعني ظهور النص في سطر واحد ، وتلك الخاصية

ضرورية لضبط اشربة التميرير في أداة النص ، لأنن أداة التميرير لا تظهر إلا في حالة السطور المتعددة .	
تحدد نوع شريط التميرير الذي سوف يظهر في أداة النص ، بدون أشربة تميرير 0-None و أفقي 1-Horizontal ، ورأسي 2-Vertical ، و كليهما 3- Both	ScrollBars
تحدد طول أو عدد الحروف التي سوف تظهرها أداة النص ، والعدد 0 يعني الحد الأقصى لعدد الحروف .	MaxLength
تضبط خلفية أداة النص .	BackColor
تضبط لون الكتابة التي سوف تظهر في أداة النص .	ForeColor
تحدد نوع جدار أداة النص ، وتأخذ قيمتين مسطح 0-None ، ومجسم 1-Fixed .	BorderStyle
المسافة التي تبعتها حافة أداة النص اليسرى عن حافة نافذة المشروع اليسرى حسب وحدة القياس .	Left
المسافة التي تبعتها حافة أداة النص العلنا عن حافة نافذة المشروع العلنا حسب وحدة القياس .	Top
عرض أداة النص حسب وحدة القياس .	Width
ارتفاع أداة النص حسب وحدة القياس .	Height
وهي تحدد ما إذا كان المستخدم سيتمكن من تعديل نص أداة النص ، وتأخذ قيمتين True لن يتمكن المستخدم من تعديل نص الأداة ، وفي هذه الحالة تكون شبيه بأداة العنوان Label بينما القيمة False تمكن المستخدم من تعديل أداة النص ، واثر تلك الخاصية يظهر في طور التشغيل فقط.	Locked
تحدد ما إذا كانت أداة النص سوف تظهر على نافذة المشروع في طور التشغيل فقط .	Visible

الأحداث Events :

ولعل خاصية النص Text هي أهم خواص أداة النص ، والمبرمج يحتاج لمعرفة متى تغيرت قيمة تلك الخاصية ، حتى يتسنى له استخدام القيمة الجديدة ، والواقع أنه يوجد حدثين يساعداننا في تنفيذ ذلك :

الحدث	الوصف
Change	وهذا الحدث يحدث كلما تغيرت قيمة النص Text الموجود في أداة النص .
LostFocus	وهذا الحدث يحدث عندما يترك المستخدم أداة النص ، ويقوم بحدث آخر مع أداة أخرى موجودة على نافذة المشروع .

وحدث التغير **Change** يحدث كثيراً فكلما ضغط المستخدم زر وكتب حرف في أداة النص يقع حدث التغير ، وهنا يتم استدعاء حدث التغير ، وتفحص قيمة خاصية النص **Text** في هذه الحالة يفيدنا بالقيمة الحالية للنص.

ويعتبر حدث **فقد التركيز LostFocus** مفيد في اختبار النص الموجود في خانة النص أو أداة النص ، ونذكر عند وضع أداة على نافذة المشروع أنه يمكننا تنشيطها بنقرها ، وبالمثل فهذا الحدث مشابه فبينما تطبيق ما يعمل في طور التشغيل ، فإنه يمكن للمستخدم أن يتعامل مع أداة واحد بحدث ومن ثم تلك هي حالة التركيز **focus** أي أن الأداة نشطة ، فعندما يكتب المستخدم في خانة النص ، فإن الأداة تكون في حالة تركيز ، وحدث **فقد التركيز LostFocus** عندما تستحوذ أداة أخرى على التركيز بأي حدث ، وهنا يمكننا أن نلاحظ تغير خاصية النص Text لملاحظة القيمة الجديدة لها .

18-1 الدرس الثاني في لغة البيسك :

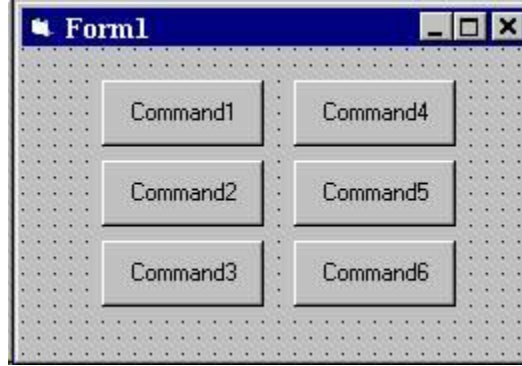
في الجزء الحالي سوف نتعرف على بعض مفاهيم لغة البيسك ، فسوف نناقش المتغيرات (الاسم Name، والنوع Type، والتصريح Declaration)

مشروع - Form Fun تصميم المشروع :

في المشروع الحالي سوف نقضي وقت ممتع مع خواص نافذة الإطار باستخدام أزرار الأوامر ، فنحن سوف نرى زر يجعل نافذة المشروع تكبر أو تنمو grow ، و آخر يجعل النافذة تتقلص shrink ، وزرين يغيرا لون نافذة المشروع ، وزر يخفي الأزرار ، وآخر يستعيد تلك الأزرار .

1-18-1 و ضع الأدوات على نافذة المشروع :

ابدأ مشروع فيجوال بيسك ، اجعل حجم نافذة المشروع تناسب ستة أزرار أوامر ، وضع ستة أزرار أوامر على نافذة المشروع ، ونحاول أن تضبط الأزرار وفق الترتيب البادي بالمصور التالي:



لعلك تنزعج من إجراءات وضع الأزرار الستة على نافذة المشروع ، ولكن لغة فيجوال بيسك توفر لنا وسيلة جيدة وهي نسخ **Copy** الزر أو أي أداة بنقر الأداة ثم لمس المفاتيح **Ctrl + C** ، ثم نلمس المفاتيح **Ctrl + V** معا للصق **Paste** الأداة وفي حالتنا هذه الأداة هي زر الأوامر ونكرر ذلك خمس مرات متتالية .

2-18-1 ضبط خواص الأدوات Set Control Properties

ننقر نافذة الخواص أو أظهرها أن كانت مخفية ، ونختار الأداة بنقرها أو باختيار اسمها من قائمة الأدوات المختارة التي تعلقو نافذة الخواص ، وخصص للأزرار الأسماء **Name** ، والعنوان **Caption** كما يلي :

قيمة الخاصية	اسم الخاصية
Name	frmFormFun
Caption	Form Fun

زر الأوامر **Command1** :

اسم الخاصية	قيمة الخاصية
cmdShrink	Name
Caption	تقليص النافذة

زر الأوامر **Command2** :

اسم الخاصية	قيمة الخاصية
cmdGrow	Name
Caption	تكبير النافذة

زر الأوامر **Command3** :

اسم الخاصية	قيمة الخاصية
cmdHide	Name
Caption	اخفي الأزرار

زر الأوامر **Command4** :

اسم الخاصية	قيمة الخاصية
cmdRed	Name
Caption	نافذة حمراء

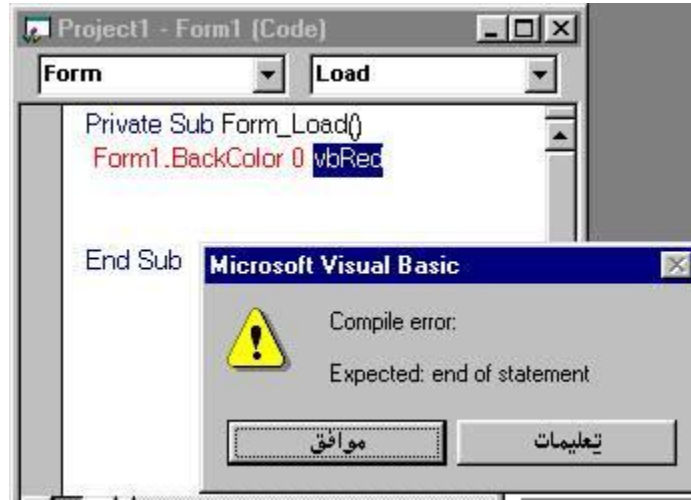
زر الأوامر **Command5** :

اسم الخاصية	قيمة الخاصية
cmdBlue	Name
Caption	نافذة زرقاء

زر الأوامر Command6 :

اسم الخاصية	قيمة الخاصية
cmdShow	Name
Caption	أظهر الأزرار
False	Visible

ويمكننا تغيير الخواص الأخرى إذا رغبت مثل خاصية الخط Font ، وعلى أي حال سوف يكون لدينا المصور التالي :



لدينا ستة أزرار أوامر اثنين لتغيير حجم نافذة المشروع ، واثنين لتغيير ألوان نافذة المشروع ، وواحد لإخفاء الأزرار ، وواحد لاستعادة الأزرار المخفية ، و نلاحظ أن الزر **أظهر الأزرار** أن قيمة خاصية فنحن لا نريده في بداية التشغيل ، وعندما نرغب في إخفاء الأزرار ، فإننا نغير False هي Visible المنظور في True ، وإذا أردنا أن نظهرهم نغير قيمة خاصية المنظور لهم إلى False قيمة خاصية المنظور لهم إلى إجراء حدث النقر للزر الذي عنوانه **أظهر الأزرار**.

1-18-3 كتابة إجراءات الأحداث Write Event Procedures:

لدينا ستة أزرار موضوعة على نافذة المشروع ، ونحتاج لكتابة شفرة لإجراء نقر **Click** كل من تلك الأزرار ، ولدينا زر يجعل نافذة المشروع تنقل ولكن التقلص لدرجة عدم وجود زر التكبير سوف يؤدي الى عدم تمكننا من تكبير النافذة ، ويمكننا تجنب ذلك بجعل النقر على نافذة المشروع ذاتها يكبر النافذة . وفي كل إجراء حدث سوف نستخدم نافذة الشفرة ، ونختار الأداة أو ننقرها نقرأ مزدوجاً ، ثم نضع المؤشر بين سطر العنوان والسطر الختامي وهو جسم الإجراء و نكتب كود أو شفرة الحدث . ولكن تأكد من اننا نكتب كل شئ حتى الملاحظات .

أولاً دعنا نكتب حدث النقر لزر الأوامر نقلص النافذة **cmdShrink_Click** وفي هذا الإجراء نقلص النافذة بمقدار 100 وحدة تويب في كل مرة ننقر فيها ذلك الزر .

Private Sub cmdShrink_Click()

تقليص نافذة المشروع'

تقليص ارتفاع نافذة المشروع بمقدار 100 وحدة تويب'

frmFormFun.Height = frmFormFun.Height - 100

تقليص عرض نافذة المشروع بمقدار 100 وحدة تويب'

frmFormFun.Width = frmFormFun.Width - 100

End Sub

وقبل أن نتعرض لبقية إجراءات الأحداث ، لننظر عن قرب الى الإجراء السابق ، فهذا الإجراء ينفذ لذي نقر الزر الذي عنوانه **قلص النافذة** ، وبالطبع فأنت نتعرف جملة الملاحظة أو التوثيق ، أما الجمل الأخرى فتغير ارتفاع وعرض نافذة المشروع ، فالجملة التي تغير الارتفاع هي :

frmFormFun.Height = frmFormFun.Height - 100

نذكر كيفية عمل مؤثر التخصيص (=) ، فالطرف الأيمن يحسب أولاً ، ومن يتم طرح (مؤثر الطرح يظهر قبل العدد 100) من ارتفاع **Height** النافذة الحالي **frmFormFun** ثم تخصص النتيجة لخاصية الارتفاع بالاستبدال مع القيمة السابقة أي الارتفاع ناقص 100 تويب ، وهنا يتقلص ارتفاع النافذة بمقدار 100 تويب.

والتعبير السابق يظهر أيضا لماذا لا نسمي مؤثر التخصيص بعلامة التساوي ، فأبي واحد يمكننا الحكم على أن الطرف الأيسر لا يمكن أن يساوي الطرف الأيسر .

والآن دعنا ننتقل لحدث نقر زر تكبير النافذة cmdGrow_Click فهذا الإجراء يزيد ارتفاع النافذة Height بمقدار 100 تويب ، وكذلك يزيد عرض النافذة Height بمقدار 100 تويب :

```
Private Sub cmdGrow_Click()
```

```
    تكبير النافذة'
```

```
    تكبير ارتفاع النافذة بمقدار 100 تويب'
```

```
    frmFormFun.Height = frmFormFun.Height + 100
```

```
    تكبير عرض النافذة بمقدار 100 تويب'
```

```
    frmFormFun.Width = frmFormFun.Width + 100
```

```
End Sub
```

و حدث النقر لزر الأوامر cmdRed_Click يغير خلفية نافذة المشروع للون الأحمر :

```
Private Sub cmdRed_Click()
```

```
    جعل لون خلفية النافذة أحمر'
```

```
    frmFormFun.BackColor = vbRed
```

```
End Sub
```

بينما حدث النقر لزر الأوامر cmdBlue_Click يغير لون خلفية نافذة المشروع للون الأزرق :

```
Private Sub cmdBlue_Click()
```

```
    جعل لون النافذة أزرق'
```

```
    frmFormFun.BackColor = vbBlue
```

```
End Sub
```

وإجراء حدث النقر للزر الأوامر cmdHide_Click (يخصص القيمة False للخاصية Visible) كل أزرار الأوامر عد الزر cmdShow الذي يجعله مرئياً .

```
Private Sub cmdHide_Click()  
cmdShow.إخفاء كل الأزرار عد الزر'  
cmdGrow.Visible = False  
cmdShrink.Visible = False  
cmdHide.Visible = False  
cmdRed.Visible = False  
cmdBlue.Visible = False  
cmdShow.إظهار الزر'  
cmdShow.Visible = True  
End Sub
```

أما إجراء حدث النقر للزر cmdShow_Click الذي يظهر كل الأزرار عد زر إخفاء الأزرار:

```
Private Sub cmdShow_Click()  
cmdShow.إظهار كل الأزرار عد'  
cmdGrow.Visible = True  
cmdShrink.Visible = True  
cmdHide.Visible = True  
cmdRed.Visible = True  
cmdBlue.Visible = True  
cmdShow.إخفاء الزر'  
cmdShow.Visible = False  
End Sub
```

وأخيراً فإجراء حدث نقر نافذة المشروع يكبرها أيضاً ، ومن ثم فله نفس أثر الزر `cmdGrow_Click`:

Private Sub Form_Click()

تكبير نافذة المشروع'

تكبير ارتفاع نافذة المشروع بمقدار 100 تويب'

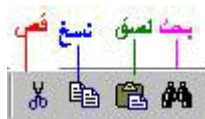
frmFormFun.Height = frmFormFun.Height + 100

تكبير عرض نافذة المشروع بمقدار 100 تويب'

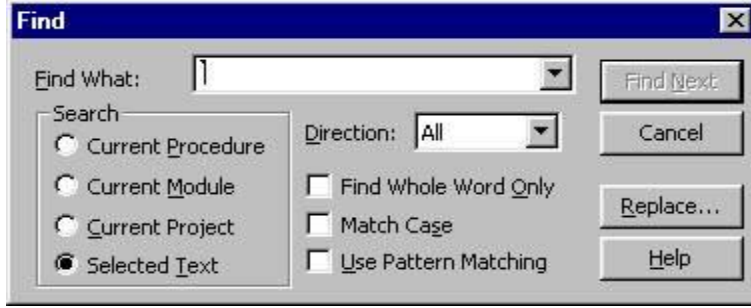
frmFormFun.Width = frmFormFun.Width + 100

End Sub

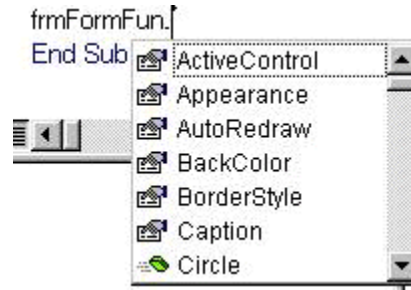
مع ملاحظة اننا نختار ت الإجراءات بطريقة جيدة ، فعندما نختار أداة النافذة سوف يظهر لنا إجراء حدث التحميل **Load** ، فنختار نا حدث النقر **Click** من قائمة الإجراءات :
ويجب أن نرى كيفية عمل تلك الإجراءات ، و انتبه لكيفية عمل خاصية المنظور **Visible** في أحداث الزرين `cmdShow` و `cmdHide` ، و نلاحظ أن هناك إجراءين متشابهين في الوظيفة وهما إجرائي الحدثين **Form_Click** و **cmdGrow_Click** ، ومن ثم نحاول أن نظل المنطقة المتكررة ونسخها ، ونقوم بلصقها في جسم إجراء الحدث المتكرر، ويمكننا أن نظل المنطقة و نختار قائمة التصحيح **Edit** ، ونختار **Copy** ثم نضع مؤشر الماوس في جسم الإجراء الثاني ، ونفتح قائمة **Edit** مرة أخرى ونختار **View** حينئذ يتم لصق الموجود بالذاكرة في ذلك الموضع ، ولكن توفر لنا فيجوال بيسك وصول مباشر لتلك الوظائف من خلال شريط الأدوات كما يلي :



فنظل المنطقة وننقر أيقونة النسخ ثم نضع المؤشر حيث نشاء ونضغط أيقونة اللصق ، وكذلك يمكننا قص المنطقة المظلمة ، بل والبحث عن كلمات من خلال أيقونة البحث ، فلدى نقرنا لها يظهر لنا المربع الحوار التالي :



بل يمكننا من خلال ذلك المربع الحوار أن نستبدل كلمة بأخرى بعد البحث عنها .
كذلك توفر لنا فيجوال بيسك طريقة عرض آلية للخواص لدي كتابتنا لاسم أداة وبعد وضعنا للنقطة تظهر لنا قائمة نختار من اسم الخاصية التي نريدها كما يلي :



19-1 أشرطة التمرير والمنطق ورسائل الإدخال والإخراج Scroll Bar

الواقع أن شريط التمرير عبارة عن مؤشر يتحرك في إطار و يكون في بدايته و نهايته سهم طرفي ، وهو إما أفقي أو رأسي ، و مؤشر الشريط يشير للمدى الذي بلغه متغير معين .



ويعتبر شريط التمرير ضروري للإشارة الى موضع نص في أداة النص ، وهو يعتبر مؤشر لبعض المتغيرات مثل درجة الحرارة مثلاً أو الارتفاع و الانخفاض وغيرها ، وأهم ما يميزه هو وجود

قيمة صغري Min يبدأ منها ولا يتجاوزها لقيمة أقل ، كما أن له قيمة كبرى لا يتجاوزها المؤشر ، كما أنه يوجد في كل طرف سهم يؤدي نقره لتحرك المؤشر مسافة محددة باتجاه ذلك الطرف ، كما يمكن سحب المؤشر ووضعه عند أي نقطة من جسم المؤشر .

خواص شريط التمرير Scroll Bar Control Properties :

اسم الخاصية	وظيفتها
Name	وهي تحدد الاسم الذي سوف يتم تسمية أداة التمرير به واقتراح أن يكون hsb للشريط الأفقي ، و vsb لشريط التمرير الرأسي .
Max	وهي تحدد الحد الأقصى الذي سوف يبلغه المؤشر ، وتقبل عدد صحيح ويتراوح بين □ 32767
Min	وهي تحدد الحد الأدنى الذي سوف يبلغه المؤشر ، وهي تقبل عدد صحيح يتراوح بين □ 32767
LargeChange	وهي تحدد الفترة التي سوف يقفزها المؤشر لدى نقر جسم أو مسار المؤشر ، وهي تقبل عدد صحيح .
SmallChange	وهي تحدد الفترة التي سوف يقفزها المؤشر لدى نقر السهم الطرفي ، وهي تقبل عدد صحيح .
Value	وهي تحدد ، وتختزن موضع المؤشر من جسم الشريط ، ويجب أن تكون محصورة بين القيمة الصغري والقيمة القصوي أو الكبرى
Left	وتحدد بعد حافة شريط التمرير اليسرى عن حافة نافذة المشروع اليسرى .
Height	وتحدد ارتفاع زر التمرير حسب وحدة القياس .
Top	وتحدد بعد حافة شريط التمرير العلوي عن حافة نافذة المشروع العلوي .
RightToLeft	وتحدد اتجاه حركة شريط التمرير ويأخذ قيمتين True وتعني اتجاه حركة المؤشر من اليمين لليسار ، والقيمة False وهي تجعل اتجاه حركة المؤشر من اليسار لليمين .

وهي تأخذ قيمتين وهما True وهنا يكون الشريط مرئياً على نافذة المشروع، أما القيمة False تجعل الشريط غير مرئي ، وتعمل في طور التشغيل .	Visible
---	---------

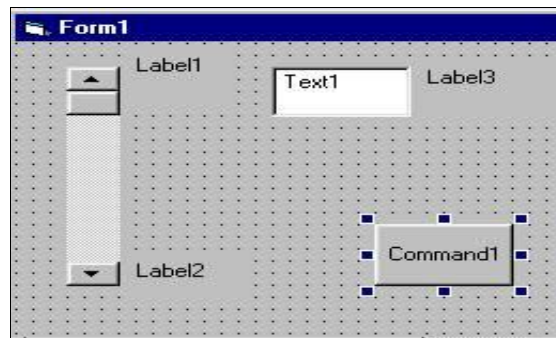
أهم الأحداث :

وصف الحدث	الحدث
وهو يقع لدى سحب مؤشر شريط التمرير باتجاه أحد الطرفين	Scroll
وهو حدث يقع لدي نقر أحد الأسهم أو سحب مؤشر شريط التمرير باتجاه أحد الطرفين .	Change

مثال :

لنفرض أننا نريد استخدام شريط تمرير رأسي لكي يشير الى درجة الحرارة المئوية والمحصورة بين 0 ، 100 ، على أن تكون القفزة الصغرى 1 ، والقفزة الكبرى 10 ، وإظهار قيمة مؤشر زر التمرير في خانة نص ، والتي سوف تخصص للإدخال ، مع تعريف خانة النص بأداة عنوان وتعريف القيمة الصغرى والقيمة الكبرى لشريط التمرير بأداة عنوان ، مع تخصيص زر للخروج .

من تأمل المسألة يتضح أننا بحاجة الى 3 أدوات عنوان ، وشريط تمرير ، وأداة نص ، و زر أوامر ، بحيث يبدو شكل نافذة المشروع كما يلي :



خصص أسماء الأدوات السابقة كما يلي :

أداة العنوان Label1

lblMax	Name
Center	Alignment
Times New Roman	Font
True	RightToLeft
100	Caption

أداة العنوان Label2

lblMin	Name
Center	Alignment
Times New Roman	Font
True	RightToLeft
0	Caption

أداة العنوان Label3

lblCent	Name
Center	Alignment
Times New Roman	Font
True	RightToLeft
درجة الحرارة	Caption

الأداة Text1

txtCent	Name
Center	Alignment
Times New Roman	Font

True	RightToLeft
	TEXT
3	MaxLength
False	Multilines

أداة زر الأوامر Command1

CmdExit	Name
Center	Alignment
Times New Roman	Font
True	RightToLeft
خروج	Caption

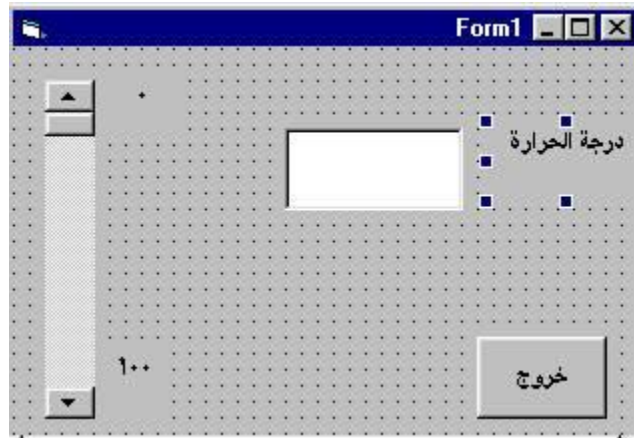
أداة شريط التمرير

الخاصية	قيمتها
Name	vsbCent
Max	100
Min	0
LargeChange	1
SmallChange	10

من الواضح أننا نحتاج الأحداث التالية للأدوات التالية

الأداة	أحداثها المطلوبة
شريط التمرير	تغير Change
أداة النص	تغير Change
زر الأوامر	نقر Click

نلاحظ أن حدث أداة شريط التمرير Scroll غير كافي لأنه يعبر عن سحب مؤشر الشريط فقط بينما حدث التغير يعبر عن التغير في موضع المؤشر بصرف النظر عن سحب المؤشر أو نقر سهم الشريط . وسوف تكون نافذة المشروع كما يلي :



في نافذة الشفرة نختار أداة النص tXTCent والحدث Change

```
Private Sub tXTCent_Change()
```

'أخذ قيمة شريط التمرير من أداة النص

```
VsbCent.Value = Val(tXTCent.Text)
```

```
End Sub
```

كما تعلم أن أداة النص تستقبل حروفية في خاصية النص Text ، بينما خاصية Value فس شريط التمرير تقبل عدد صحيح ، لذا يجب تغيير العدد الحرفي الذي ندخله في أداة النص الى عدد بالدالة Val .

ومن قائمة الأدوات نختار أداة شريط التمرير VsbCent ، ونختار الحدث Change ونكتب فيه الشفرة التالية :

```
Private Sub VsbCent_Change()
```

'أخذ نص أداة النص من قيمة شريط التمرير

```
tXTCent.Text = Str(VsbCent.Value)
```

```
End Sub
```

كما تعلم أن أداة النص تقبل حروفية String بينما الخاصية Value من شريط التمرير تقبل عدد صحيح ، وبالتالي يجب تحويل قيمة شريط التمرير الى حروفية بالدالة Str .

ومن قائمة الأدوات أيضاً نختار أداة زر الأوامر Cmdexit ونختار الحدث Click ونكتب فيه الشفرة التالية :

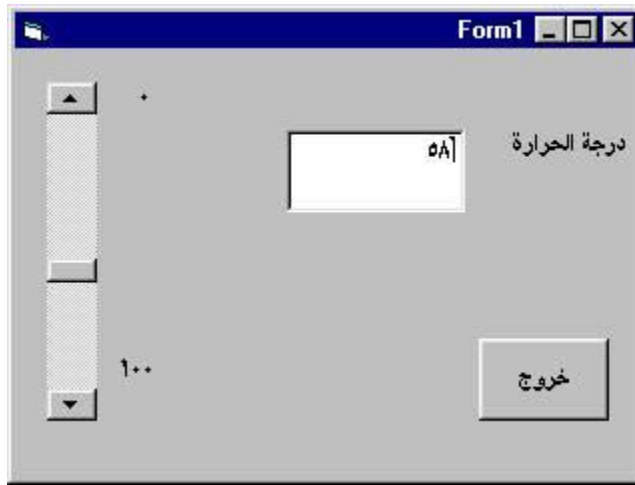
```
Private Sub VsbCent_Change()
```

```
'أخذ نص أداة النص من قيمة شريط التمرير
```

```
tXTCent.Text = Str(VsbCent.Value)
```

```
End Sub
```

ولدي تشغيل البرنامج يكون لدينا شكل النافذة التالية :



نحاول أن نغير القيمة الصغرى والقيمة الكبرى ، وتمرس على استخدامهما مع الخاصية Value وغير موضع زر التمرير في أماكن عدة و نلاحظ الخاصيتين Top ,Left ، وخير حجم شريط التمرير و نلاحظ أثر ذلك على الخاصيتين Width و Height ، ونحاول أن نضيف زر يجعل شريط التمرير يختفي ، وأضف زر أوامر آخر يستعيد شريط التمرير المخفي (Visible) .

20-1 زر الاختيار Option Button:

يستخدم زر Option Button عندما يكون لدينا خيار إما صحيح أو خطأ مثلاً الأسئلة التي تتطلب نعم أو لا ، صواب ، وخطأ ، الخ .



أهم الخواص :

الخاصية	اسمها	وظيفتها
Caption	العنوان	تخزن العنوان الذي سوف يظهر بجوار الزر.
Value	قيمة الخاصية	وهي تخزن قيمتين الأولى True وتعني أن الزر تم نقره ، وهنا تظهر بداخله نقطة سوداء ، والقيمة False وتعني أن الزر لم يتم نقره ، ولا تظهر به أي نقاط .
RightToLeft	من اليمين لليسا	يجعل اتجاه النص من اليمين لليسا وهو مفيد للنص العربي
Left	البعد الأيسر	يحدد البعد عن حافة نافذة المشروع اليسرى بوحدة القياس النشطة .
Top	البعد عن القمة	يحدد البعد عن حافة نافذة المشروع العليا بوحدة القياس النشطة .
Height	الارتفاع	يحدد ارتفاع زر الاختيار بوحدة القياس النشطة .
Width	العرض	يحدد عرض زر الاختيار بوحدة القياس النشطة .
Visible	منظور	يحدد مدى تمكن المستخدم من مشاهدة الزر في طور التشغيل من عدمه True = منظور و False = مخفي
Enabled	تمكين	يحدد مدى تمكن المستخدم من نقر الزر في طور التشغيل من عدمه True = تمكين و False = عدم تمكين

وهناك خواص أخرى مثل الخاصية Font و BackColor و هما يحدد الخط ولون الخلفية ، أما لون الخط فتحدده الخاصية ForeColor .

مثال :

ببدا مشروع فيجوال بيسك جديد ، ونضع عليه الأدوات التاليتين :

الأداة	اسمها	خصائصها
نافذة المشروع	Form1	Caption التحكم في النافذة بأزر الاختيار
		BackColor &H00008000&
		Name Form1
		RightToLeft True
زر أوامر	Command1	Caption خروج
		Name CmdExit
زر اختيار	Option1	Name optFormgreen
		RightToLeft True
		BackColor &H00008000& (ننقر اللون)
		Alginment 1-Right justify
		Caption أخضر
زر اختيار	Option1	Name OptFormBlue
		RightToLeft True
		BackColor &H00FF0000&
		Alginment 1-Right justify
		Caption أخضر

وسيبدو شكل نافذة المشروع كما يلي :



وفي حدث النقر لزر الأوامر نكتب الشفرة التالية :

```
Private Sub CmdExit_Click()  
ينهي البرنامج'  
End  
End Sub
```

و في حدث النقر لزر الاختيار OptFormBlue نكتب الشفرة الموجودة في جسم الإجراء :

```
Private Sub OptFormBlue_Click()  
يجعل خلفية نافذة المشروع زرقاء'  
If OptFormBlue.Value = True Then Form1.BackColor = vbBlue  
End Sub
```

و في حدث النقر لزر الاختيار optFormgreen نكتب الشفرة الموجودة في جسم الإجراء :

```
Private Sub optFormgreen_Click()  
يجعل خلفية نافذة المشروع خضراء'  
If optFormgreen.Value = True Then Form1.BackColor = vbGreen  
End Sub
```



21-1 خانة التحقق Check Box :

خانة التحقق تعمل بطريقة مشابهة لزر الاختيار Option Box ولكن يمكننا أن نختار أكثر من خانة تحقق في نفس الحاوية ، وأقرب نمط لذلك هي الأسئلة التي تحمل صفات متعددة .
وفيما يلي شكل أداة خانة التحقق على نافذة الأدوات وعند وضعها على نافذة المشروع:



وكما نرى أن أداة خانة التحقق عبارة عن مربع يحوي عنوان ، وللأداة عدة خصائص شبيهة بزر الاختيار ، وعند نقر الأداة في طور التشغيل تظهر علامة صح مما يعني اختيار الأداة .

خصائص أداة خانة التحقق :

الخاصية	قيمتها	وظيفتها
Caption	نص	تخزن عنوان أداة خانة التحقق ، ويجب أن تكون حروفية .
BackColor	عدد سادس عشري	تحدد لون الخلفية ، ويمكننا نقر اللون المناسب .
Enabled	False-True	تخزن إحدى القيمتين فالقيمة False تتيح للمستخدم نقر الأداة في طور التشغيل ، والقيمة True تجعل الأداة باهتة ، وهذا مفيد في حالة توافق الخيار مع خيار آخر .
Visible	False- True	تخزن قيمة منطقية ، فالقيمة True تتيح للمستخدم رؤية الأداة وبالتالي التعامل معها ومع وظائفها ، بينما القيمة False تخفي الأداة ، وهذا مفيد في حالة عدم توافق الخيار مع خيار آخر ، ولكن يجب توفير أسلوب لاستعادتها .
Value	Checked- Unchecked - Grayed	تعتبر تلك الخاصية من أهم الخواص ، فهي تخزن قيمة تعبر عن نقر الأداة Checked أو عدم نقرها Unchecked أما القيمة Grayed فتظهر الأداة معلمة ولكن رمادية اللون ، وعند نقرها في طور التشغيل يختفي اللون الرمادي والنقر مرة أخرى يعلمها ، فهي نصف معلمه .

1- 22 اداة الصورة المحدودة :PictureBox

قدمت فيجوال بيسك أداة لتحميل صور متنوعة وتتمتع بعدة خصائص ولكن أهمها الخاصية Picture ، وتلك الخاصية تمكنك من تحميل صورة مختزنة في ملف وتلك الصورة يمكن أن تكون من عدة أنواع :

Bitmaps
Gif
Metafile
JPEG
ICON

وهي كافية لعرض معظم الصور التي قد تجدها في بيئة ويندوز ويمكننا أن تحمل الصورة في طور التصميم وفي طور التشغيل . شكل الأداة على

نافذة المشروع	نافذة الأدوات
	

هذه الأداة تتيح لنا اضافة الصور إلى تطبيقاتنا. ومعالجة هذه الصور وتحريكها
خصائص هذه الأداة:

Name	خاصية الإسم وهي من الخصائص المعروفة لدينا
Align	وضع الصورة هل تكون في يمين الإطار أم اليسار أم ...
Appearance	طريقة الظهور
AutoRedraw	إعادة الرسم التلقائي
AutoSize	تحجيم الإطار تلقائيا بحجم الصورة الموضوعه فيه
BackColor	اللون الخلفي للإطار
BorderStyle	تحديد الشكل الخارجي للإطار
Enabled	هل الصورة فعالة أم لا
Height	خاصية ارتفاع الإطار

Left	خاصية بعد الإطار عن أقصى يسار الفورم
MouseIcon	شكل أيقونة الماوس من الأشكال التلقائية..
MousePointer	اختيار شكل مؤشر الماوس من أيقونة خارجية
Picture	اختيار صورة ووضعها داخل إطار الأداة
ToolTipText	نص المساعدة الذي يظهر لو توقفت بالماوس أعلى الصورة لمدة قصيرة
Top	خاصية بعد الصورة عن أعلى الفورم
Visible	هل الصورة ظاهرة أم مخفية
Width	خاصية عرض الصورة

1-2-1 تحميل الصور أثناء التشغيل:

من المعروف أن هناك عدد من التطبيقات التي تعتمد على الصور الخارجية. فلو كنا نريد أن نصنع متصفح للصور مثلاً مثل AcdSee فسيكون علينا تحميل الصور من ملفات خارجية إلى تطبيقنا. ويمكن ذلك مع الأداة PictureBox عن طريق الأمر LoadPicture..

وصيغة الأمر LoadPicture تكتب كالتالي:

```
Picture1.Picture = LoadPicture(PicturePath)
```

والـ Picture Path لابد وأن يكون المسار كاملاً

1-22-2 قلب الصورة:

يمكننا نسخ صورة و لصقها مقلوبة باستخدام هذه الاداه فقط ضع 2 Picture Box واحدة بها الصورة و واحدة التي سينقل اليها الصورة مقلوبة, و لقلبه نستخدم هذه الاكواد:

(1) الوضع الطبيعي للنسخ:

```
Private Sub Command1_Click()
```

```
Picture2.PaintPicture Picture1.Picture, 0, 0, _
```

```
Picture1.Width, Picture1.Height, 0, 0, _
```

```
Picture1.Width, Picture1.Height, vbSrcCopy
End Sub
```

(2) الوضع الافقي:

```
Private Sub Command2_Click()
Picture2.PaintPicture Picture1.Picture, 0, 0, _
Picture1.Width, Picture1.Height, Picture1.Width, _
0, -Picture1.Width, Picture1.Height, vbSrcCopy
End Sub
```

(3) الوضع الرأسى:

```
Private Sub Command3_Click()
Picture2.PaintPicture Picture1.Picture, 0, 0, _
Picture1.Width, Picture1.Height, 0, Picture1.Height, _
Picture1.Width, -Picture1.Height, vbSrcCopy
End Sub
```

(4) قلب الصورة:

```
Private Sub Command4_Click()
Picture2.PaintPicture Picture1.Picture, 0, 0, _
Picture1.Width, Picture1.Height, Picture1.Width, _
Picture1.Height, -Picture1.Width, -Picture1.Height, vbSrcCopy
End Sub
```

23-1 اداة الصور Image Box

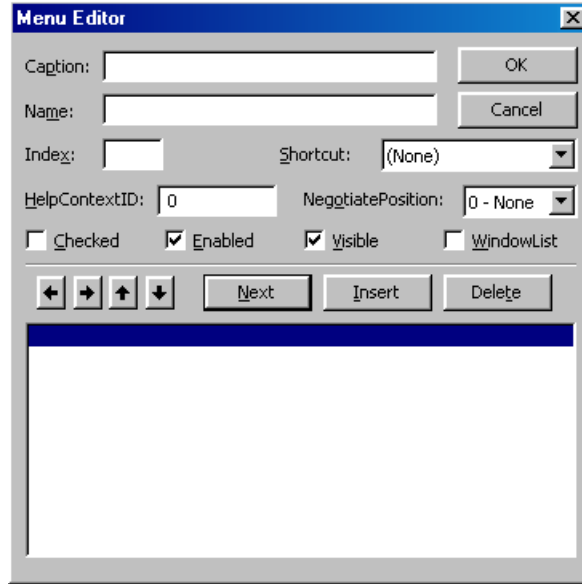
هي الأداة الثانية التي نتيج لنا وضع الصور داخل تطبيقاتنا وهي تتميز عن الأداة السابقة في بعض الخصائص وتعب عنها في بعض الخصائص الأخرى، أهم خصائص الأداة :

Name	خاصية الإسم
BorderStyle	شكل الإطار الخارجي
Enabled	هل الأداة فعالة أم لا
Height	ارتفاع الأداة
Left	بعد الأداة عن يسار الفورم
MouseIcon	شكل الماوس
MousePointer	مؤشر الماوس
Picture	تحميل صورة من ملف خارجي في الأداة
Stretch	تحجيم الصورة بحجم الأداة
ToolTipText	النص الذي يظهر عند التوقف بالماوس فوق الأداة مدة قصيرة
Top	خاصية بعد الأداة عن قمة الفورم
Visible	هل الأداة ظاهرة أم مخفية
Width	خاصية عرض الأداة

وتتميز هذه الأداة بخاصية Stretch والتي لا تتوفر في الأداة PictureBox وهي من الخواص يمكن تحميل الصور أثناء التشغيل كما في اداة PictureBox او باستخدام CommonDialog.

24-1 القوائم Menue

لاضافة قائمة الى برنامجنا فيمكننا الضغط على Menu Editor في شريط الادوات, او اختيار قائمة Menue Editor Tools, سيظهر صندوق هكذا:



محتوياته:

بجانب كلمة Caption يوضع ما يكتب في القائمة.

بجانب كلمة Name نكتب اسم القائمة.

بجانب Shortcut نقوم بأختيار الاختصار مثل Ctrl + N ... الخ.

Checked معناها ان التعامل مع القائمة سيكون بالعلامات.

Enabled و هي اذا كانت القائمة متاحة ام لا.

Visible و يقصد بها ان اذا كانت القائمة ظاهرة ام لا.

اذا اردنا ان نفتح قائمة من قائمة فرعية فنقوم بالضغط على السهم → بجانب Next لفتح قائمة جديدة من القائمة التي قبلها و للعودة الي القائمة الرئيسية نضغط علي ←.

إذا أردنا أن يوضع خط تحت حرف ما في الكلمة أي أن عند الضغط على Alt ثم هذا الحرف لتعمل فنقوم بوضع & قبل الحرف المراد وضع خط تحته. وإذا أردنا وضع فاصل فنضع - في خاصية Caption.

لإنشاء قائمة عن طريق الشفرة (البرمجة) نتبع الآتي:

ننشئ قائمة فرعية باسم ما وليكن Mnu, نغير Index الي 0 ثم نكتب هذه الشفرة في زر الإضافة:

```
Dim index As Integer
```

```
index = Mnu.Count
```

```
Load Mnu (index)
```

```
Mnu(index).Caption = "قائمة جديدة"
```

```
Mnu(index).Visible = True
```

قائمة جديدة هو اسم القوائم الجديدة التي ستنشأ، ويمكننا تغييره كما يمكننا كتابة Text1.Text بدلا من الاسم و وضع TextBox لتنشأ القائمة باسم النص.

الفصل الثاني

المتغيرات والعمليات الرياضية

1-2 المتغيرات Variables

كل أجهزة الكمبيوتر تتعامل مع معلومات سواء أكانت عددية أو نصوص ، أو ألوان أو صور ، وبالتالي نحتاج برامج الكمبيوتر أماكن لحفظ و تخزين تلك المعلومات أثناء العمل أو التشغيل ، وقد شاهدنا أنواع القيم التي تخزنها خواص أدوات الفيچوال بيسك مثل حجم الأداة ، وظهورها - مسطحة - مجسمة- وموضعها على نافذة المشروع ، كذلك الألوان . ولكن خواص الأدوات ليست كافية لحفظ و تخزين كل المعلومات التي يحتاجها المشروع . ولاختزان المعلومات في أشياء أخرى غير الخواص، نحتاج لأشياء تسمى متغيرات variables ، وتسمى متغيرات لأن المعلومات التي تخزنها يمكن أن تتغير أثناء تشغيل البرنامج ، فالمتغيرات أشياء تتحرك خلالها المعلومات ، ولكن يجب علينا اتباع قواعد معينة في كتابة المتغيرات ، وهي تشبه ما تعرضنا له في الخواص . المتغير عبارة عن مكان غير مرئي تحفظ به بعض البيانات المؤقتة، و سرعان ما تفقد قيمتها و مساحتها عند اغلاق البرنامج.

انواع المتغيرات:

رقمية: و هي متغيرات تقبل ارقاما فقط و تتيح عملية ضرب و الطرح ..الخ.

حرفية: و هي متغيرات تقبل سلاسل حرفية مهما كان محتواها.

عملة: و هي متغيرات التي تقبل ارقام العملات.

تاريخ - وقت: و هي متغيرات تحمل قيم التاريخ و الوقت.

متغيرات متخصصة: و هي متغيرات تخصصها انت بنفسك عن طريق الامر Format.

متغيرات Variant: و هي متغيرات تقبل كل الصور السابقة و لكن تشغل مساحة كبيرة من الذاكرة.

أسماء المتغيرات Variable Names

يجب علينا تسمية كل متغير في برنامجنا وفقاً للقواعد التالية :

- يجب أن لا يزيد طول الاسم عن 40 حرف .
- يمكننا فقط استخدام الحروف والأرقام والشرطة السفلية (_) .underscore
- يجب أن يكون أول حرف في الاسم هو حرف وليس رقم .
- لا يمكننا استخدام أحد الكلمات -الأوامر- المحفوظة لفيجوال بيسك مثل كلمة PRINT ، أو كلمة Form أو كلمة BEEP.

وأكثر قاعدة مهمة هي أن يكون لاسم المتغير دلالة ومعنى لما يخزنه من معلومات ، مثل Score للدرجات ، TotalPrice للثمن الاجمالي حتى تتمكن من تحديد المعلومات التي يخزنها بفحص قيمة المتغير ، وهذا ضروري عند البحث عن الأخطاء المنطقية .

2-2 أنواع المتغيرات Variable Types

نحن نحتاج لمعرفة نوع Type المعلومات المخزنة بكل متغير ، وهي نفس الأنواع المستخدمة في الخواص ، فلدينا عدد صحيح integer ، وعدد صحيح طويل long integer ، وقيمة منطقية Boolean ، وقيمة حرفية string ، وتوجد أنواع أخرى تجدها في ملف التعليمات .
وهنا سوف نتعرض لنوع المتغيرات المسمى مفرد single ، فنحن نعرف أنه حتى الآن فإن كل المشروعات التي تعرضنا لها استخدمنا أعداد صحيحة أو أعداد أكبر ، ونحن نعرف الكثير عن الأعداد الحقيقية في الرياضيات بما في ذلك الأعداد العشرية ، فالنوع المفرد هو عدد يحوي علامة عشرية ، وبلغة الكمبيوتر نسمي ذلك بعدد ذو علامة عائمة floating point number وفيما يلي أمثلة لأعداد من النوع المفرد أو المنفرد :

8.00 -1234.980 4.404040

والمتغيرات يمكن أن تظهر في جمل التخصيص كما بالصيغة التالية :

VariableName = NewValue

مثل :

MyName="صادق"

MyName="خالد المياح"

FirstRoot=n/m

فقط نضع المتغير المفرد في الطرف الأيسر ثم علامة التخصيص (=) ثم نضع في الطرف الأيسر أي قيمة مشروعة ، ونذكر أن الطرف الأيسر **NewValue** يحسب أولاً ثم يخصص للمتغير **VariableName** ، والجدير بالذكر انه يجب أن تكون القيمة **NewValue** من نفس نوع المتغير **VariableName** ، وهو ما شاهدناه في الخواص .

التصريح عن المتغيرات **Declaring Variables**:

عندما نسمي المتغير ونحدد نوعه يجب أن نعلم فيجوال بيسك ، فنحن نحتاج للتصريح أو الإعلان **declare** عن متغيراتنا ، ونحن لم نعلن خواص الأدوات لأن فيجوال بيسك تعلمهم ، والجملة المستخدمة للإعلان أو التصريح عن أسماء المتغيرات هي

Dim VariableName As Type

مثال:

Dim MyName As String

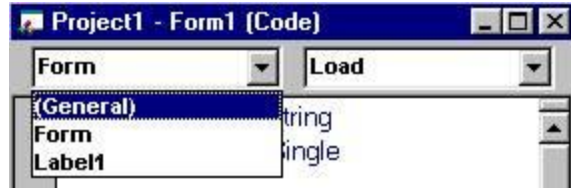
Dim FirstRoot As Single

Dim n As Integer

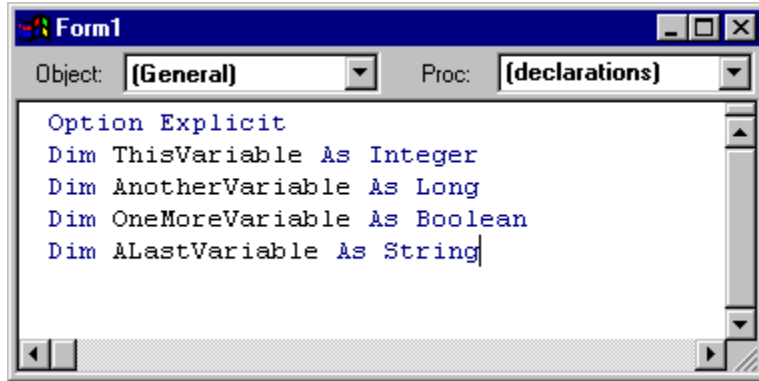
Dim m As Integer

ونحن نحتاج للتصريح عن كل متغيراتنا المستخدمة في مشروعنا ، وربما يبدو هذا يستغرق جهد كبير ولكنه هام ، فإعلان عن المتغيرات يجعل البرمجة أسهل ، ويقلل من احتمالية حدوث الأخطاء ، ويسهل من تعديل البرنامج لاحقاً .

نبدأ مشروع فيجوال بيسك و نستحضر نافذة الشفرة ، وننقر قائمة الأهداف ، وهنا نجد كلمة **General** في القائمة وكلمة **Form** ، نختار كلمة **General** ومن ثم تبدو نافذة الشفرة كما يلي :

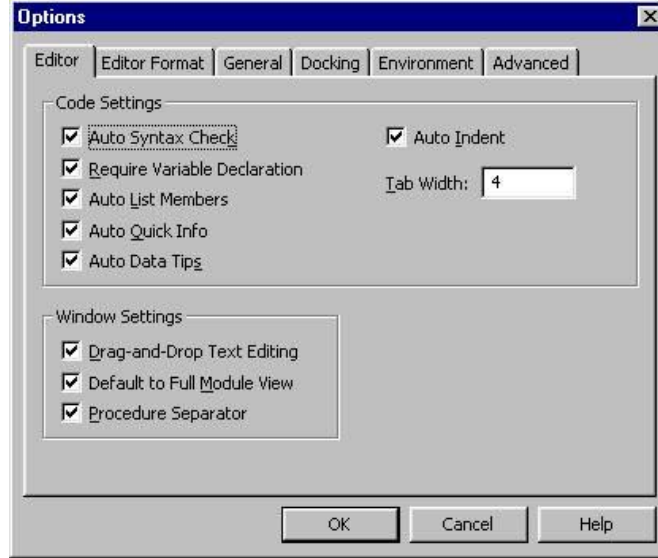


ونقر كلمة General سوف يجعل شكل نافذة الشفرة كما يلي :

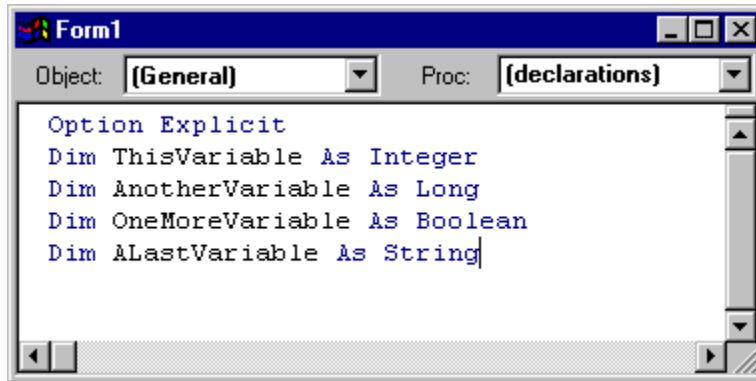


وتلك المنطقة التي تكون في أول جزء في البرنامج هي منطقة الإعلان عن المتغيرات وتسمى منطقة التعاريف العامة **general declarations** ، وكل متغير يتم الإعلان عنه في تلك المنطقة يمكن استخدامه في أي إجراء أو أي جزء من البرنامج .

ويجب أن تسبق الإعلان عن تلك المتغيرات بكلمة **Option Explicit** - قد لا يكون ذلك ضرورياً بالرغم من أهميته - ويمكن أن تضبطه من قائمة **Tools** ، الخيار **Options** ثم انقر **Editor** ، وننقر الخيار **RequireVariable Declaration**



فهذا السطر يخبر فيجوال بيسك بأننا ينبغي أن نعلن عن كل متغيراتنا ، وإن كان ليس من الضروري التصريح عن كل المتغيرات ومن الآن فصاعداً يجب أن تصرح عن متغيرات أو تعلن عنها ، وسوف نشاهد كيف يساعد هذا في تسهيل مهام البرمجة ، نحاول أن نكتب بعض المتغيرات في منطقة التعاريف ، وفيما يلي أمثلة :



```

Option Explicit
Dim ThisVariable As Integer
Dim AnotherVariable As Long
Dim OneMoreVariable As Boolean
Dim ALastVariable As String

```

وفائدة الجملة option explicit هي أنه عند تشغيل البرنامج ووجد متغير لم يتم الإعلان عنه يخرج لنا رسالة خطأ تفيد بأن المتغير لم يعلن عنه:



وهذا مفيد في حالة المتغيرات الكبيرة .

3-2 العمليات الحسابية Arithmetic Operators

من المهام التي يؤديها الكمبيوتر بشكل جيد العمليات الحسابية ، فيمكنه جمع add ، وطرح subtract ، وضرب multiply ، وقسمة divide الأعداد بسرعة ، والواقع أن لدينا 7 عمليات حسابية arithmetic operators في لغة البيسك .

فعملية الجمع **Addition** تتم بإشارة الجمع (+) ، وعملية الطرح **subtraction** تتم باستخدام إشارة الطرح (-) ، وفيما يلي أمثلة :

المؤثر	مثال	النتيجة
Addition	7 + 2	9
Addition	3 + 8	11
Subtraction	6 - 4	2

وعمليّة الضرب **Multiplication** تتم باستخدام العلامة (*) ، و عمليّة القسمة **division** تتم باستخدام الرمز (/) وفيما يلي أمثلة :

العملية	مثال	النتيجة
الضرب	Multiplication	32
الضرب	Multiplication	24
القسمة	Division	6
القسمة	Division	7

ومن المؤكد اننا استخدمنا تلك العمليات و علمنا خواصها (الإبدال والدمج والتوزيع... الخ) ، كما يوجد في لغة البيسك ثلاثة عمليات أخرى :

و عمليّة الأس **exponentiation** يمثل بالرمز (^) وهو الرمز العلوي للمفتاح الذي يحمل الرقم 6 ويمكننا استدعاء الرمز بالضغط على المفتاح <Shift + 6> ، ويستخدم الأس عندما نرغب في ضرب عدد في نفسه عدة مرات أو حساب قيمة جذر عدد ، فإذا كان لدينا عددين صحيحين A,B فإن A^B تعني A أس B أو اننا نضرب A في نفسها B مرة ، وفيما يلي أمثلة :

مثال	Result
5^2	25
2^4	16
3^3	27

كما يوجد عملية يرتبط بعملية القسمة للعدد الصحيح **integer division** ، وهو يشبه مؤثر القسمة فيما عدا أنه يهمل الكسور ، بينما يعمل مؤثر المقياس **modulus** بطريقة عكسية لعملية دالة العدد الصحيح ، فهو يهمل العدد الصحيح الناتج من القسمة ويخرج باقي القسمة ، أي أن العمليتان يقتسما خارج القسمة

،فالجزء الذي يتجاهله أحدهما يخرج الآخر ، وبينما يرمز لمؤثر العدد الصحيح بالرمز **INT** أو العلامة \ يرمز لمؤثر المقياس بالرمز **Mod** وفيما يلي أمثلة توضيحية :

العملية	مثال	نتيجة القسمة	نتيجة العملية
دالة العدد الصحيح	7 \ 2	3 Remainder 1	3
دالة العدد الصحيح	23 \ 10	2 Remainder 3	2
دالة العدد الصحيح	18 \ 3	6 Remainder 0	6
دالة العدد الصحيح	25 \ 5	5 Remainder 0	5
دالة المقياس	7 Mod 4	1 Remainder 3	3
دالة المقياس	14 Mod 3	4 Remainder 2	2
دالة المقياس	25 Mod 5	5 Remainder 0	0

نلاحظ الفرق بين علامة القسمة / وعلامة قسمة العدد الصحيح \

لغة البيسك تراعي ما يعرف بأولوية الترتيب ، وبالتالي تحدد ترتيب إجراء العمليات التي سوف تجريها طبقاً لذلك ، وهذا هو الترتيب :

1. الأس (^)
2. الضرب (*) و القسمة (/)
3. دالة العدد الصحيح (\)
4. المقياس (Mod)
5. الجمع (+) و الطرح (-)

لذا ففي أي جملة تخصيص يتم حساب الأس أولاً ثم الضرب والقسمة ، ثم قسمة العدد الصحيح ، والمقياس وأخيراً الجمع والطرح ، فمثلاً (4 * 3 + 7) يتم حساب عملية الضرب فنجدها 12 ثم نجمع على الناتج 7 ليصبح الناتج النهائي 19 .

وإذا كان لعمليتين نفس الأولوية مثل (الجمع والطرح) أو الضرب والقسمة ، فإن العملية تتم من اليسار لليمين في جملة التخصيص فمثلاً :

$$24 / 2 * 3$$

لدينا قسمة وضرب ولهما نفس درجة الأولوية ، ولكن تحسب هكذا :

$$24 / 2 * 3$$

$$12 * 3$$

$$36$$

ولكن يمكننا أن نجري عملية الضرب أولاً بدلاً من القسمة بو نضع أقواس حول عملية الضرب وهنا يتم ما بداخل الأقواس أولاً ويصبح له الأولوية .

example as:

$$24 / (2 * 3)$$

$$24 / 6$$

$$4$$

ولكن نلاحظ أن كل قوس يتكون جزأين فنقص أحدهما يسبب رسالة خطأ ، وهي من النوع التركيبي.
مثلاً :

$$d=a+(b*c)$$

هناك بعض الرموز التي مرت يجب ان تعرفها قبل كتابة الكود.

<	أصغر من
>	أكبر من
<>	لا يساوي
<=	أصغر من أو يساوي
>=	أكبر من أو يساوي
+	زائد للجمع
-	ناقص للطرح أو سالب
*	في للضرب

على للقسمة	/
القسمة بدون كسور	\
الأس	^
الباقى من القسمة	Mod

إذا اردت وضع علامات مثل Cos و Tan, فيمكنك استخدام الدوال الرياضية.

4-2 دوال رياضية

الدوال الرياضية Mathematics Functions :

الدالة Abs : ترجع القيمة المطلقة لأي عدد وترجعه من نفس نوع البيانات المعطى للدالة والمقصود بالقيمة المطلقة هي قيمة العدد بدون إشارة فالقيمة المطلقة ل (-13) مثلا هي (13) وهكذا, فمثلا لو كتبنا الشفرة التالي..

Number=Abs(-45.6)

Text1.Text = Number

فإن نتيجة تنفيذ الدالة هي Number=45.6

ولاحظ أن القيمة المدخلة للدالة لا بد أن تكون عدد أو تعبير عددي فإذا كانت القيمة المدخلة للدالة Null ستكون النتيجة Null وإذا كانت القيمة المدخلة للدالة متغير فارغ أو لم يتم تعيين قيمة له ستكون النتيجة صفر.

الدالة Sqr : تستخدم هذه الدالة في تحديد الجذر التربيعي لرقم معين وتأخذ الصورة العامة التالية.

Number=Sqr(25)

Text1.Text = Number

فإن نتيجة تنفيذ الدالة هي Number=5 .

الدالة Log: تستخدم هذه الدالة في تحديد قيمة اللوغاريتم العشري لرقم وتأخذ الصورة العامة التالية :

Number=Log (20)

Text1.Text = Number

فإن نتيجة تنفيذ الدالة هي $MyNumber=2.9957327$..

الدالة Int: وتستخدم هذه الدالة لحساب الجزء الصحيح فقط من رقم يشتمل على أرقام صحيحة وعشرية أو بعبارة آخر لحذف الأرقام العشرية الموجودة بعد العلامة العشرية بدون تقريب وتأخذ الصورة التالية:

Number=Int (332.54)

Text1.Text = Number

فإن نتيجة تنفيذ الدالة هي $MyNumber=332$

الدالة Atn: تستخدم هذه الدالة في حساب مقلوب ظل الزاوية "ظنا" Atan للرقم الذي تشتمل عليه مقدار بالتقدير الدائري وتأخذ الصورة العامة التالية:

MyNumber=Atn (رقم)

Text1.Text = MyNumber

الدالة Tan: تستخدم هذه الدالة في تحديد قيمة ظل زاوية معينة وتأخذ الصورة العامة التالية :

MyNumber=Tan (رقم)

Text1.Text = My Number

الدالة Cos: وتستخدم هذه الدالة في تحديد قيمة جيب تمام الزاوية معينة وتأخذ الصورة العامة التالية:

MyNumber=Cos (رقم)

Text1.Text = MyNumber

الدالة Sin: تستخدم هذه الدالة في تحديد قيمة جيب زاوية معينة وتأخذ الصورة العامة التالية:

Number=Sin (رقم)

Text1.Text = Number

5-2 التعامل مع التاريخ

زيادة ايام او شهور على التاريخ الاصلي نستخدم هذا الشفرة:

Label1.Caption = DateAdd(Interval, Number, Date)

حيث تكون Interval هي المراد زيادته سواء كان ايام "D" او شهور "M", و Number هو رقم الايام او الشهور او السنين المراد زيادتها, و Date هو التاريخ الحالي. و كما يتبين ان النتيجة ستظهر في Label.

مثال:

Date1 = DateAdd("M", 3, Date)

المقارنة بين تاريخين:

Label1.Caption = DateDiff(Interval, "Date1", "Date2")

Date1 هي التاريخ الاول, و Date2 هي التاريخ الثاني, اما Interval فهي المراد ظهور النتيجة به من شهور "M" و ايام "D", و لظهورها بالسنين الكتب هذا الشفرة:

Label1.Caption = DateDiff("m", "1/8/2015", "24/9/2015") \ 12

اظهار الساعة و التاريخ سهل جدا وهو بكتابة هذا الشفرة في التايمر Timer الذي وضعته:

```
Private Sub Timer1_Timer()  
Label1.Caption = Time  
Label2.Caption = Date  
End Sub
```

الساعة و التاريخ يؤخذان من ساعة و تاريخ System.

6-2 السلاسل الحرفية

السلاسل الحرفية Strings هي الصورة التي يتم بها تخزين النصوص في ذاكرة الحاسوب. جميع الدوال التي تتعامل مع النصوص يتم تمرير نص إليها في صورة متغير حرفي String. يمكن دمج متغيرين حرفين كما بهذا الشفرة:

```
Dim String1 As String  
Dim String2 As String  
String1 = "Haider"  
String2 = "Mohammed"  
Label1.Caption = String1 & String2
```

وهكذا اعلنا عن متغيرين من نوع حرفي و عرفناهم, و قمت بوضع Label و اظهار قيمة String1 و String2 به.

1-6-2 الدالتين Str و Val

من المهام الشائعة في مشروعات فيجوال بيسك أن يدخل المستخدم أرقام لتنفيذ بعض العمليات على تلك الأعداد ، فقد تستخدم المؤثرات الحسابية لحساب النتيجة وأداة العنوان Label لعرض النتائج ولكن لدينا مشكلتين هما :

□ أن العمليات الحسابية تعمل فقط مع الأعداد (مثل المتغيرات الصحيحة والخواص الصحيحة) ولكن القيم التي يدخلها المستخدم من خلال أداة النص تكون حرفية String وبالتالي لا يمكننا أن تضرب المتغيرات الحرفية أو القيم الحرفية .

□ أن نتيجة إجراء العمليات الحسابية تكون أعداد بينما أداة العنوان أو نص تقبل قيم حرفية ، فلا يمكننا أن نخزن قيم عددين في متغيرات حرفية مباشرة .

لذا فنحن نحتاج لحل تلك المشكلتين ، والحل موجود في دوال الفيچوال بيسك ، فيمكننا أن نحول القيم العددية الى حرفية بالدالة STR ، ويمكننا استرجاع تلك الأعداد التي تحولت لحروف بالدالة Val .
وهنا يمكننا أخذ الأرقام من أداة النص و تحويلها لأعداد بالدالة Val ثم تجري عليها العمليات ، وتأخذ الناتج وتحوله لحرفية ، وتضعه مباشرة في خاصية العنوان Caption في أداة العنوان Label أو أداة النص TextBox . وصيغة الدالة العامة هي :

FunctionValue = FunctionName(ArgumentList)

مثلاً :

StrValue = STR(NumValue)

NumValue =Val (StrValue)

حيث **FunctionName** هو اسم الدالة ، و **ArgumentList** هي القيمة المراد تحويلها من والى (حرفي و عدد) ، ويجب أن يكون مدخل الدالة مناسب لها ، فالدالة **Val** مدخلها عدد حرفي ، والدالة **Str** مدخلها عدد.
فالدالة Val تحول عدد حرفي لعدد ، والدالة Str تحول عدد لعدد حرفي .

YourNumber = Val(YourString)

YourNumber = Val("70")

فجملة التخصيص خصصت للمتغير العددي **YourNumber** العدد 70 لذلك يمكن إجراء العمليات الحسابية عليه

YourString = Str(70)

ولكن المتغير **YourString** الآن يخزن عدد حرفي هو 70 ، لا يمكننا إجراء العمليات الحسابية ، ولكنه يعالج كحروف او رمز .

والواقع يجب أن تألف تحويل الأعداد لحرفية والعكس باستخدام الدالتين Val و Str فهما استخدامات شائعة في مشروعات الفيجوال بيسك .

2-6-2 وصل وربط الحروفيات String Concatenation

كما ذكرنا أنفاً فالأعداد الحرفية لا يمكن جمعها أو ضربها و لكنها قد تعطي نتيجة غريبة عند الجمع :

```
Print Str (7) + Str (0)
```

7 0

```
Print Str (0) + Str (7)
```

0 7

فالواقع أن ما فعلته وهو عملية وصل لحرفية بحرفية أخرى ، وكما نلاحظ فتوجد مسافة بينهما . ويمكننا استخدام مؤثر الربط أو الوصل (&) حتى لا تتشوش بالعملية + الذي يشير لعملية الجمع الحسابية ، وهو الرمز الذي يعلو العدد 7 وتحصل عليه 7 + Shift مثال :

```
NewString = "Visual " & "Basic"
```

```
Print NewString
```

الناتج :

```
Visual Basic
```

نلاحظ أن المسافة المتروكة بعد حرف I هي التي تفصل بين الكلمتين ، فإذا نفذنا السطر التالي :

```
Text1.Text = "سير " & "الرجل " & "اسرع"
```

نحصل على

```
سيرالرجلاسرع
```

3-6-2 تغيير حالة الاحرف

- تحويل الاحرف الي احرف كبيرة:

نضع TextBox, و زر امر و نكتب هذا الشفرة:

```
Text1.Text = UCase(Left(Text1.Text, Len(Text1.Text)))
```

- تحويل الاحرف الى احرف صغيرة:

ضع TextBox, و زر امر و اكتب هذا الشفرة:

```
Text1.Text = LCase(Left(Text1.Text, Len(Text1.Text)))
```

- الحرف الاول كبير و الباقي صغير:

ضع TextBox, و زر امر و اكتب هذا الشفرة:

```
X = Text1.Text
```

```
Y = UCase(Left(X, 1))
```

```
Z = LCase(Right(X, Len(X) - 1))
```

```
Text1.Text = Y & Z & g
```

- عكس الاحرف:

نضع TextBox, و زر امر و نكتب هذا الشفرة:

```
Text1.Text = StrReverse(Text1.Text)
```

4-6-2 دوال نصية

الدالة Len: ترجع طول نص معين وتستخدم كما يلي:

```
Dim Stl As Integer
```

```
Stl = Len(String1)
```

الدالة Trim: تمحو المسافات من بداية ونهاية النص, وتستخدم كما يلي:

```
Result = Trim(String1)
```

الدالة LTrim: تمحو المسافات من بداية النص . وتستخدم كما يلي:

```
Result = LTrim(String1)
```

الدالة RTrim: تحو المسافات من نهاية النص . وتستخدم كما يلي:

```
Result = Rtrim(String1)
```

الدالة Left: ترجع عدد معين من الأحرف من بداية النص, ويستخدم كما هو موضح:

```
Result = Left(String1, 1)
```

الدالة Right: ترجع عدد معين من الأحرف من نهاية النص وتستخدم كما هو موضح:

```
Result = Right(String1, 1)
```

الدالة Mid: ترجع عدد معين من الأحرف من وسط النص بداية من حرف محدد وتستخدم كما يلي:

```
Result = Mid (String1, 2, 1)
```

الدالة chr(): تعطى الحرف المقابل للكواد AscII وتستخدم كالاتي:

```
Result = Chr(13)
```

الدالة ASC(): تعطى كود ASCII المقابل لحرف معين وتستخدم كالاتي:

```
Dim AscCode
```

```
AscCode = Asc("E")
```

الدالة Str () : لتحويل المتغير الرقمي إلى صيغة نصية . وتستخدم كما يلي:

```
Result = Str("5")
```

الدالة Val () : لتحويل المتغير النصي إلى صيغة رقمية وتستخدم كما يلي:

```
Dim Number
```

```
Number = Val(String1)
```

جميع ارقام هذه الدوال قابلة للتغيير, اما بالنسبة لكلمة String1 فهي اي متغير حرفي من نوع String و Result الكائن المراد ظهور النتيجة به وليكن Label مثلا.

7-2 صناديق الإدخال والإخراج

صندوق الإدخال Input message :

هل نذكر الجملة INPUT في البيسك ، إنها تغيرت الآن ، وأصبحت لها صيغة مختلفة وشكل أفضل فصيغته هي InputBox:

Variable =InputBox ("الرسالة")

Answer =InputBox ("من فضلك أدخل إجابتك")

وبالطبع فالمتغير Answer سوف يتقبل حروفه ، ومن ثم إذا أردت أن يكون المدخل أعداد فاستخدم الدالة Val .



في المربع الأبيض يدخل المستخدم إجابته ثم ينقر الزر OK

صندوق الرسائل MsgBox :

لنفرض اننا نريد أن تخرج رسالة للمستخدم ، هنا فيجوال بييسك قدمت لنا أمر يمكننا من إخراج الرسائل التي نريدها من خلال نافذة جميلة نستخدم MsgBox

MsgBox ("Thank you")

الصيغة العامة لصندوق الرسائل هو:

(عنوان مربع الحوار ,الازرار ,والرموز, الرسالة المطلوبة) MsgBox .

الرسالة المطلوبة : هي الجملة التي تريد عرضها على المستخدم.

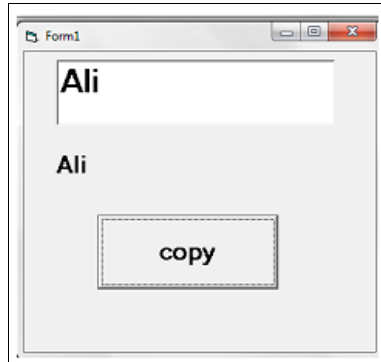
عنوان مربع الحوار : هو عنوان شريط عنوان مربع الحوار.

الازرار والرموز: وهو يحدد نوعية الازرار التي ترغب في عرضها والرموز. والرموز والازرار يوضحها الجدول التالي فيظهر لنا ان كل زر او مجموعة او رمز حجزت له قيمة ثابتة بحيث يتم استخدام هذه القيم منفردة أو يتم مزجها بعملية الجمع لعرض زر معين او مجموعة ازرار مع رمز.

القيمة	المعنى المقابل
0	Ok(موافق)
1	Ok(موافق) و cancel(الغاء الامر)
2	Abort(احباط) و Retry (اعادة المحاولة) Ignore(تجاهل)
3	Yes (نعم) و No (لا) Cancel (الغاء الامر)
4	Yes (نعم) و No (لا)
5	Retry (اعادة المحاولة) Ignore(تجاهل)
16	عرض رمز قف
32	عرض رمز علامة الاستفهام
48	عرض رمز علامة التعجب
64	عرض رمز علامة معلومات

8-2 مجموعة أمثلة

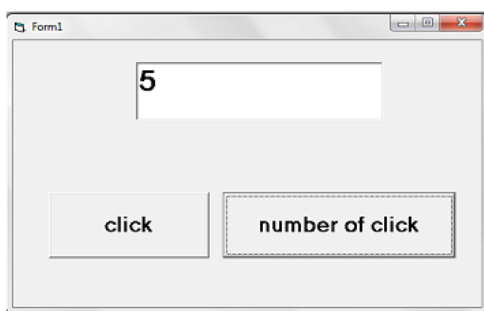
مثال: اكتب برنامج يقوم بنسخ الاسم المكتوب بداخل text الى Label



الحل:

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


مثال: اكتب برنامج لحساب عدد النقرات على زر معين.



الحل:

```
Dim x As Integer  
Private Sub Command1_Click()  
x = x + 1  
End Sub  
  
Private Sub Command2_Click()  
Text1.Text = x  
End Sub
```

مثال :- كتيب برنامج لتنفيذ حاسبة بسيطة.



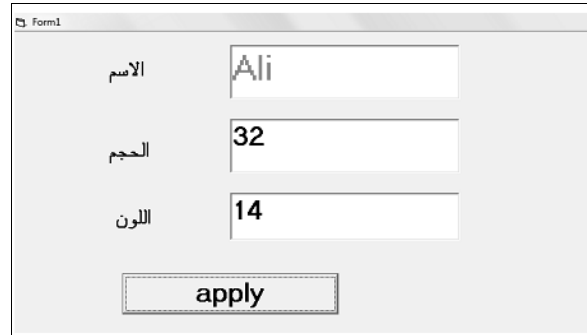
```
Private Sub Command1_Click()  
Text3 = Val(Text1.Text) + Val(Text2.Text)  
End Sub
```

```
Private Sub Command2_Click()  
Text3 = Val(Text1.Text) - Val(Text2.Text)  
End Sub
```

```
Private Sub Command3_Click()  
Text3 = Val(Text1.Text) * Val(Text2.Text)  
End Sub
```

```
Private Sub Command4_Click()  
Text3 = Val(Text1.Text) / Val(Text2.Text)  
End Sub
```

مثال :- اكتب برنامج لتغيير حجم ولون خط داخل $\langle text \rangle$



The screenshot shows a Windows form titled 'Form1'. It contains three text boxes with labels in Arabic: 'الاسم' (Name) containing 'Ali', 'الحجم' (Size) containing '32', and 'اللون' (Color) containing '14'. Below the text boxes is a button labeled 'apply'.

```
Private Sub Command1_Click()  
Text1.FontSize = Val(Text2.Text)  
Text1.ForeColor = QBColor(Val(Text3.Text))  
End Sub
```


مثال :- اكتب برنامج لاجاد باقي قسمة اي عددين.



The screenshot shows a Windows form titled 'Form1'. It contains three text boxes with labels in Arabic: 'العدد الاول' (First Number) containing '26', 'العدد الثاني' (Second Number) containing '8', and 'باقي القسمة' (Remainder) containing '2'. Below the text boxes is a button labeled 'apply'.

```
Private Sub Command1_Click()  
Text3 = Val(Text1.Text) Mod Val(Text2.Text)  
End Sub
```

مثال :- اكتب برنامج لايجاد ناتج رفع اي عدد لاس معين.



```
Private Sub Command1_Click()  
Text3 = Val(Text1.Text) ^ Val(Text2.Text)  
End Sub
```

مثال :- اكتب برنامج لايجاد مقلوب عدد من ثلاث مراتب.



```
Private Sub Command1_Click()  
x = Val(Text1.Text)  
X1 = x \ 100  
x = x - X1 * 100  
X2 = x \ 10  
x3 = x Mod 10  
Text2 = X1 + 10 * X2 + 100 * x3  
End Sub
```

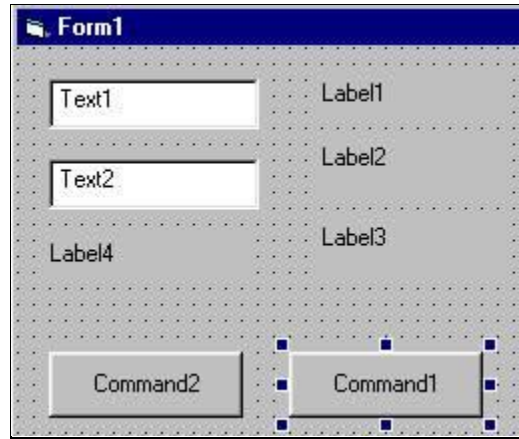
9-2 مشروع - حساب الادخار

تصميم المشروع

في المشروع الحالي سوف نبني مشروع لحساب المدخرات ، وسوف نخصص موضع للإيداع ، وموضع لعدد الأسابيع، ثم نخصص موضع لعرض اجمالي المبلغ وسوف نحتاج لأداة النص لإدخال المعلومات ، وأداة العنوان لعرض اجمالي المبلغ ، وأداة زر الأوامر ، لتنفيذ الحساب والخروج من البرنامج .

و نضع الأدوات على نافذة المشروع :

ابدأ مشروع فيجوال بيسك جديد ، وضع على نافذة المشروع 2 أداة نص ، 4 أداة عنوان ، 2 زر أوامر ، ورتبهم بحيث يظهر مظهر نافذة المشروع كما يلي :



ولا نستخدم خاصية النسخ ، نضع الأدوات بنقر أدواتها في نافذة المشروع ، لكل أداة على نحو منفصل

ضبط خواص الأدوات :

نضبط خواص الأدوات السابقة على النحو التالي :

أداة نافذة المشروع Form1:

اسم الخاصية	قيمة الخاصية
Name	frmSavings

حساب الإدخار	Caption
1-Fixed Single	BorderStyle
Arabic Transparent	Font
Alignment	Right
True	RightToLeft

أداة النص Text1:

اسم الخاصية	قيمة الخاصية
Name	txtDeposit
Text	[Blank]
Font	Arabic Transparent
Font Size	10
Center	Alignment
RightToLeft	True

أداة النص Text2:

اسم الخاصية	قيمة الخاصية
Name	txtWeeks
Text	[Blank]
Font	Arial
Font Size	10
Font	Arabic Transparent
Center	Alignment
RightToLeft	True

أداة العنوان Label1 :

اسم الخاصية	قيمة الخاصية
Name	lblDepositHeading
Caption	Weekly Deposit
Font	Arabic Transparent
Font Size	10
Right	Alignment
RightToLeft	True

أداة العنوان Label2 :

اسم الخاصية	قيمة الخاصية
Name	lblWeeksHeading
Caption	Number of Weeks
Font	Arial
Font Size	10
Font	Arabic Transparent
Right	Alignment
RightToLeft	True

أداة العنوان Label3 :

اسم الخاصية	قيمة الخاصية
Name	lblTotalHeading
Caption	Total Savings
Font Size	10
Font	Arabic Transparent
Right	Alignment
RightToLeft	True

أداة العنوان Label4 :

اسم الخاصية	قيمة الخاصية
Name	lblTotal
Caption	[Blank]
Font	Arabic Transparent
Font Size	10
BackColor	White
BorderStyle	1-Fixed Single
Right	Alignment
RightToLeft	True

ونستخدم الخاصية BackColor بجعل الخلفية بيضاء ، والخاصية BorderStyle باختيار القيمة 1- Fixed تصبح أداة العنوان Label مشابهة لأداة النص Text من حيث منظر الرؤيا .

زر الأوامر Command1 :

اسم الخاصية	قيمة الخاصية
Name	cmdExit
Caption	خروج
Font	Arabic Transparent
Font Size	14
RightToLeft	True

زر الأوامر Command2 :

اسم الخاصية	قيمة الخاصية
Name	cmdCompute
Caption	احسب
Font	Arabic Transparent
Font Size	14
RightToLeft	True

نلاحظ أن خاصية Text في أداتي النص يجب تركها خالية ، وكذلك الخاصية Caption للأداة Label4 ، وهذا يتم بتعليم القيمة الحالية ولمس مفتاح الحذف Delete ، ونراعي ضبط حجم خط اللغة العربية على خط أكبر من الخط الإنجليزي بـ 2 ، وعندما ننتهي مما سبق سيكون شكل نافذة المشروع كما يلي :

The screenshot shows a window titled "حساب المدخرات" (Savings Calculation) with the following components:

- txtDeposit**: Input field for weekly deposit, labeled "الإيداع الأسبوعي".
- txtWeeks**: Input field for number of weeks, labeled "عدد الأسابيع".
- lblTotal**: Output field for total savings, labeled "إجمالي المدخرات".
- cmdCompute**: Button labeled "احسب" (Calculate).
- cmdExit**: Button labeled "خروج" (Exit).

كتابة إجراءات الأحداث :

في المشروع الحالي يدخل المستخدم الإيداع الأسبوعي للمدخرات - بافتراض أن قيمة الإيداع ثابتة- في خانة الإيداع الأسبوعي ، ثم يدخل عدد الأسابيع التي سوف يستمر في الإيداع فيها ثم ينقر الزر احسب فيخرج الكمبيوتر إجمالي المبلغ أما خانة إجمالي المدخرات ، وعلى ذلك فالحدث الأساسي هو نقر زر

الأوامر المعنون بـ **احسب** - نذكر أن لدينا حدث نقر آخر هو حدث الخروج - ونحن نحتاج لمتغيرات من النوع الصحيح لتخزن كل من الإيداع **Deposit** ، وعدد الأسابيع **Weeks** ، و إجمالي المبلغ **Total** .
نفتح نافذة الشفرة وفي منطقة الإعلانات أو التصريح ، نصرح أو نعلن عن متغيرائنا كما يلي :

Option Explicit

Dim Deposit As Integer

Dim Weeks As Integer

Dim Total As Integer

ونحدث النقر للزر احسب **cmdCompute_Click** يتضمن الخطوات التالية :

1. تحويل قيمة الإيداع من أداة النص (**txtDeposit.Text**) لعدد ونخزن القيمة في المتغير **Deposit** .

2. تحويل عدد الأسابيع من أداة النص (**txtWeeks.Text**) لعدد ونخزن القيمة في المتغير **Weeks** .

3. ضرب قيمة الإيداع الأسبوع **Deposit** في عدد الأسابيع **Weeks** ونخزن الناتج في المتغير **Total** .

4. تحويل القيمة المختزنة في المتغير **Total** الى حرفية و نوصلها أو نربطها بكلمة "الف دينار" ونضع الناتج في خاصية العنوان **Caption** لأداة العنوان **lblTotal**

وفيما يلي توضيح بالشفرة لتلك الخطوات :

Private Sub cmdCompute_Click()

الحصول على الإيداع'

Deposit = Val(txtDeposit.Text)

الحصول على عدد الأسابيع'

Weeks = Val(txtWeeks.Text)

حساب إجمالي المدخرات'

Total = Deposit * Weeks

عرض اجمالي المدخرات'

lblTotal.Caption = Str(Total) + "الف دينار"

End Sub

نلاحظ سهولة ترجمة الخطوات السابقة لشفرة الفيچوال بيسك، فهي تستدعي فقط الاهتمام بالتفاصيل ، وبالتحديد استخدام الدالتين Str و Val ، ووظيفتهما تحويل عدد لحروفية ، وتحويل عدد حرفي الى عدد على الترتيب.

ونحدث نقر زر الخروج وإنها البرنامج **cmdExit_Click** يبدو أكثر بساطة فهو يحتوي على سطر واحد هو الأمر (End) .

Private Sub cmdExit_Click()

End

End Sub

و الآن نحفظ مشروعنا بملفيه وشغله :

تشغيل المشروع Run:

نشغل المشروع وننقر أداة النص المقابلة لـ الإيداع الأسبوعي ونكتب قيمة ، وأمام عدد الأسابيع نكتب قيمة ، ثم ننقر الزر احسب ، وهنا سوف نحصل على اجمالي المبلغ ، نتأكد من صحة الناتج بتجريب قيم نتعرف إجابتها 2 ، 4 والناتج (8) الخ ، ولدي التأكد من سلامة البرنامج ننقر زر الخروج .

والمشروع كما نرى غير معقد ، فهو فقط يحسب ناتج ضرب عددين ، ولكنه يوضح خطوات بناء مشروع الفيچوال بيسك ، ولكنه أتاح خبرة بكيفية قراءة القيم ، وتحويل الأعداد لحروف والعكس ، وجملة التخصيص ، وتنسيق الأزرار والنتائج .

تطوير المشروع :

لنفرض اننا أردنا استثمار المبلغ الذي أودعناه في البنك ، والبنك سوف يقدم لنا ربح ، فكما نرى فحساب المدخرات تجاهل الأرباح ، نحاول أن نضمن مشروعنا حساب الأرباح ، وسوف نقدم لنا الخطوات التالية ولكن ليست بالتفصيل :

- نعرف متغير اسمه الربح **Interest** لاختزان قيمة الربح السنوي ، وقيمة الربح السنوي ستتضمن أعداد عشرية لذا نجعل نوع المتغير **Single** .
- نضف أداة نص أخرى لإدخال نسبة الربح باسم **txtInterest** .
- نضف أداة عنوان **Label** لتعريف أداة النص التي أضفناها ونضع في عنوانها نسبة الربح .
- نحول الشفرة لتستخدم المتغير **Interest** لحساب الاجمالي **Total** ، بالصيغة :

$$\text{Interest} = \text{Val}(\text{txtInterest.Text})$$

ويكون قيمة الاجمالي **Total** :

$$\text{Total} = 5200 * (\text{Deposit} * ((1 + \text{Interest} / 5200) ^ \text{Weeks} - 1) / \text{Interest})$$

والآن نشغل البرنامج ، وندخل قيم إيداع ، وعدد الأسابيع ، ونسبة الربح مثل 10 ، عدد الأسابيع 20 ونسبة الربح 6.5 والمفروض أن تحصل على الاجمالي 202 الف دينار ، نلاحظ أن البرنامج السابق كان سيحسب لنا الاجمالي 200 الف دينار.

ولعلنا الآن نذكر توصيتنا بالتصريح عن جميع متغيرات ، وحتى يتضح ذلك نضع في المشروع السابق أي متغير لم نعلن عنه .

الفصل الثالث

ادوات المنطق و الشرط و العدادات والمصفوفات

1-3 المنطق والشرط

في اللغة العربية ندرس ما يعرف بالشرط ، ويتكون مما يلي :

إذا كان الجو ملبدا بالغيوم ، فإنه من المحتمل مطره .

إذا تسمي	أداة الشرط
الجو ملبدا بالغيوم	فعل الشرط
من المحتمل مطره	جواب الشرط

وتقول الرياضيات أن هناك أنواع للجمله فهناك جمل منطقية أي يمكن الحكم عليه بالصواب أو الخطأ ، وأخري تكون مفتوحة لا نستطيع الحكم عليها بالصواب والخطأ ، لنسببة الحكم عليها . ولا شك أننا نهتم هنا بالجمل التي تأخذ الشكل الرياضي أي تحوي تعبير رياضي لأنه من السهل تمثيلها أو التعبير في الفيچوال بيسك ، بل نستطيع أن تعبر بقوة عن الأحداث المنطقية باستخدام المؤثرات المنطقية :

2-3 الاداة If ..Then..Elseif ..Endif

اداة If من اشهر ادوات فيچوال بيسك, و تستخدم في الشروط فمعناها ان اذا حدث شيء فسيحدث هذا الشيء, و ان لم يحدث سيحدث شيء اخر, ولكن كيف نكتب هذا في طريقة البرنامج؟

تتكون هذه الاداة من عدة اشياء و هي كالآتي:

- 1) IF: و هي اول شيء يكتب في الابعاز و معناها ان اذا تحقق شيء معين.
- 2) Then: تكتب بعد الحدث الذي يكتب بعد If و معناها ان اذا تحقق الشيء (الحدث بعد If) فسيحقق شيء آخر.
- 3) Else: و هي اذا لم يتحقق الحدث الذي بعد If يتحقق شيء لآخر, و هي ليست درورية اي لا تستخدم في كل الحالات.

(4) ElseIf: تكون حدث جديد فهي مثل If العادية, ولكن بدلا من ان نقوم بكتابة ايعاز جديد به If و End If نقوم بكتابة ElseIf و تكلمة اليعاز.

(5) End If: و تكتب في نهاية اليعاز و هي مهمة جدا و تكتب في كل اليعاز.

و الان لنقوم بتطبيق بسيط علي ما سبق بكتابة اليعاز كامل.

```
Private Sub Command1_Click()
```

```
If Text1.Text = "A" Then
```

```
Lable1.caption = "A"
```

```
ElseIf Text1.Text = "B" Then
```

```
Lable1.caption = "B"
```

```
Else
```

```
Lable1.caption = " "
```

```
End If
```

```
End Sub
```

3-3 أداة الشرط أو OR :

إذا كان لدينا قطعة عملتي على أحد أوجهها صورة ، وعلى الوجه الآخر كتابة ، وأطحت بالقطعتين في الهواء ، فهل لنا أن تخمن عندما تستقر القطعتين على الأرض ما السطح الظاهر لنا ؟ ما لأوجه الأخرى التي من المحتمل أن تراها ؟

قطعة العملة الثانية	قطعة العملة الأولى
صورة	صورة
كتابة	صورة
صورة	كتابة
كتابة	كتابة

وأداة الربط أو OR تعني بالحكم على تعبيرين -أو أكثر- فمثلاً :

نتيجة الحكم		أحمد في القاهرة أو في الإسكندرية
= صواب	(صواب أو صواب)	$5 > 2$ أو $3 > 2$
= صواب	(صواب أو خطأ)	$3 > 2$ أو $3 = 2$
= خطأ	(خطأ أو صواب)	$3 < 2$ أو $7 > 5$
= خطأ	(خطأ أو خطأ)	$3 < 2$ أو $3 = 2$

فأداة الشرط المنطقية تعطي حكم صواب طالما أن أحد الجملتين صحيح ، وتعطي حكم خطأ إذا كان كلتا الجملتين خطأ .

وفيما يلي جدول يوضح قيمة الصحة الجملة المركبة لأداة الربط المنطقية OR

س	ص	س Or ص
صواب	صواب	صواب
صواب	خطأ	صواب
خطأ	صواب	خطأ
خطأ	خطأ	خطأ

وفي لغة البيسك :

X=5 : y=10

IF X<= 6 OR Y>6 THEN ANSER\$="نعم"

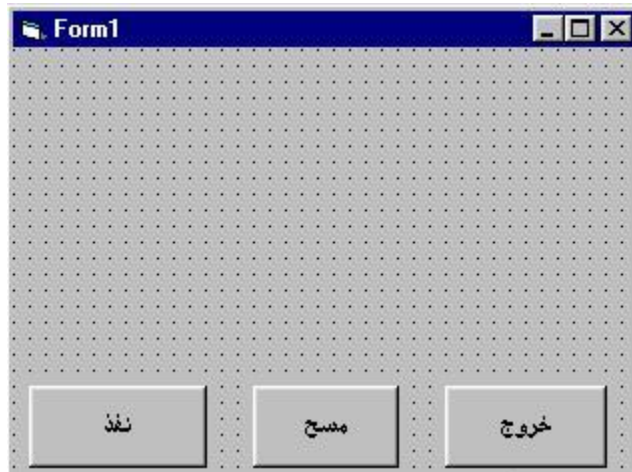
مثال :

ابداً مشروع جديد في فيجوال بيسك وضع على نافذة المشروع ثلاث أزرار أوامر حسب الجدول التالي :

Command1	Command1	Command1
----------	----------	----------

CmdRun	Name	CmdCls	Name	CmdExit	Name
نفذ	Caption	امسح	Caption	خروج	Caption

حتى تبدو شكل نافذة المشروع كما يلي :



ففي منطقة التعريف نكتب :

Option Explicit

Dim x As Integer

Dim y As Integer

Dim answer As String

وفي حدث النقر للزر CmdExit نكتب :

```
Private Sub CmdExit_Click()
```

```
End
```

```
End Sub
```

وفي حدث النقر للزر CmdCls نكتب :

```
Private Sub CmdCls_Click()
```

```
Form1.Cls
```

End Sub

لعلك نذكر من نظام التشغيل الأمر CLS فهو يمسح الشاشة ، كذلك فهو في لغة البيسك يمسح ، ولكنه هنا مقترن بنافذة المشروع Form1 لذا فهو يمسح نافذة المشروع . وفي حدث النقر على الزر CmdRun نكتب الشفرة التالية :

```
Private Sub cmdrun_Click()
```

```
x = 5: y = 10
```

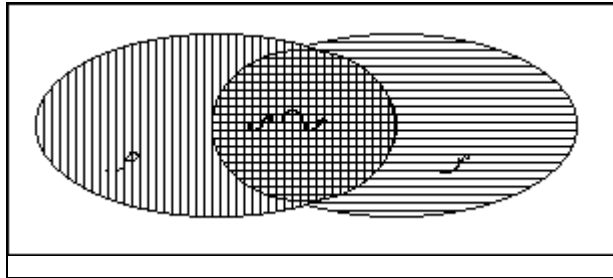
```
If x <= 6 Or y > 6 Then answer = "نعم"
```

```
Form1.Print answer
```

```
End Sub
```

4-3 أداة الربط المنطقية و And:

تعتبر أداة الربط المنطقية "و" and من الأدوات المنطقية الهامة ، وتعطي نتيجة صحيحة في حالة صواب التعبيرين فقط



ص and س

فبلغة المجموعات هي عبارة عن تقاطع المجموعتين ، بينما أداة الربط أو هي اتحاد المجموعتين ، فتعتبر العبارة المركبة من أداة الربط المنطقية "و" and صحيحة في حالة صحة التعبيرين فقط ولا تعتبر صحيحة في الاحتمالات الأخرى .

والجدول التالي يوضح قيمة الصحة للعبارة أو الجملة المركبة من أداة الربط المنطقية "و" and:

س	ص	س and ص
صواب	صواب	صواب
صواب	خطأ	خطأ

خطأ	صواب	خطأ
خطأ	خطأ	خطأ

أمثلة من البيسك :

$x = 5: y = 10$

If ($x \leq 6$ and $y > 6$) Then answer = "نعم"

Form1.Print answer

4-3 أداة النفي Not

تغير أداة الربط المنطقية ليس Not من حالة الجملة من الصواب الى الخطأ والعكس

Not $2 > 4$	True
Not $3 < 2$	True
Not $5 < 10$	False

ففي منطقة التعريفات نكتب :

Option Explicit

Dim x As Integer

Dim y As Integer

Dim answer As String

وفي حدث النقر للزر CmdExit نكتب :

Private Sub CmdExit_Click()

End

End Sub

وفي حدث النقر للزر CmdCls نكتب :

Private Sub CmdCls_Click()

Form1.Cls

End Sub

لعلك تتذكر من نظام التشغيل الأمر CLS فهو يمسخ الشاشة ، كذلك فهو في لغة البيسك يمسخ ، ولكنه هنا مقترن بنافذة المشروع Form1 لذا فهو يمسخ نافذة المشروع . وفي حدث النقر على الزر CmdRun نكتب الشفرة التالية :

```
Private Sub cmdrun_Click()  
x = 5: y = 10  
If Not(x >= 6 and y < 6) Then answer = "نعم"  
Form1.Print answer  
End Sub
```

5-3 العدادات والدورات Loops

1-5-3 الاداة Do-Loop

الايغاز Do.. Loop تستخدم في تكرار شيء ما الى ان يحدث شيء ما , و تتكون من:

- (1) Do : و هي اول شيء يكتب لفعل شيء.
- (2) Loop Until : تكتب بعد الحدث الذي يكتب بعد Do و معناها ان هذا الشيء سيتم تكريره حتي يحدث شيء ما . يتم كتابة الشفرة كالاتي:

```
Private Sub Command1_Click()  
Score.Caption = "0"  
Do  
Score.Caption = Score.Caption + 1  
Loop Until Score.Caption = "10"  
End Sub
```

ففي هذا الكود عندنا Label اسمه Score , يقوم البرنامج باضافة 1 الى الرقم حتى تصل الي الرقم عشرة..

2-5-3 الايغاز For-Next

هذه القاعدة من اهم قواعد الحركة التكرارية و تكتب كالاتي:

```
Private Sub Command1_Click()
```

```
For i = 1 To 10 Step 2
```

```
Print i
```

```
Next i
```

```
End Sub
```

كما نرى من الشفرة السابقة انه ليس عندنا سوى كائن form و زر امر, و عند الضغط عليه, يتم طباعة ارقام علي form من 1 الي عشرة ولكن خطوتان خطوتان اي نكتب الارقام: 1, 3, 5, 7, 9.

شرح الشفرة:

في السطر الاول كتبت ان i هي الارقام من 1 الي 10 عن طريق كتابة For i = 1 To 10, اما Step 2 فهي خطوة العد, و في السطر Print i يقوم البرنامج بطباعة ما حدث, و Next i هي نهاية الشفرة.

3-5-3 الاداة While-Wend

هذا الابعاز يستخدم في فعل شيء معين اثناء حدوث حدث معين, و عند انتهاء هذا الشيء ينتهي الحدث, و تتكون القاعدة من: While , Wend

مثال: يتم كتابة الشفرة كالآتي:

```
Private Sub Command1_Click()
```

```
Label1.Caption = "0"
```

```
While label1.Caption < 10
```

```
Label1.Caption = Label1.Caption + 1
```

```
Wend
```

```
End Sub
```

و في هذا المثال عندنا Label , و طالما الرقم الموجود به اصغر من 10 فيقوم البرنامج بزيادة 1 الى ان نصل الي الرقم عشرة.

4-5-3 الاداة Select Case

تصلح عبارة الشرط if إذا كان جواب الشرط عبارة عن احتمالين أو ثلاثة أما إذا كنا نتوقع عند نقيمننا لشرط معين احتمالات كثيرة فمن الأفضل أن نستخدم عبارة Select Case وتكون صيغتها العامة ما يلي:

تبدأ العبارة بـ Select Case يليها اسم المتغير او التعبير الذي سيتم اختباره.
تأتي بعد ذلك الاحتمالات Case بعد كل منها احدى قيم المتغير الذي ستتم مقارنته ثم يعقبها التعليمات التي ستنفذ إذا كان الشرط صحيحًا أو كان المتغير بهذه القيمة.
واخيرًا يأتي Case else ومعناها إذا كان المتغير لا يساوي أيًا من القيم السابقة أو إذا لم يكن الشرط صحيحًا فإن التعليمات التي تلي Else هي التي تنفذ .

مثال: ضع Textbox1 و زر امر command1 و نكتب به هذا الشفرة:

```
Private Sub Command1_Click
```

```
    Select Case Text1.Text
```

```
        Case "A"
```

```
            MsgBox "Your Character A"
```

```
        Case "B"
```

```
            MsgBox " Your Character B "
```

```
        Case Else
```

```
            MsgBox "Wrong!"
```

End Select

End Sub

كما يتبين من الشفرة انه عند الضغط علي الزر command1, اذا كانت $A = \text{Textbox1.text}$, فسيقوم البرنامج باظهار رسالة, و اذا كانت B, فيقوم البرنامج بارسال رسالة مختلفة و اذا لم تكن هذا او ذاك فسيظهر رسالة اخرى.

3-6 دالة العدد العشوائي RND :

دالة العدد العشوائي تولد أعداد عشوائياً أقل من 1 بمعنى غير مرتبة ، فلا يمكننا أن نتعرف العدد ، وحتى يتضح لنا وظيفة الدالة RND ابدأ مشروع فيجوال بيسك جديد وضع زر أوامر وضع وننقره ونضع الشفرة التالية فيه :

Option Explicit

Dim z As Single

Private Sub Command1_Click()

z = RND (1)

Print z

End Sub

شغل البرنامج نجد الأرقام كما يلي :



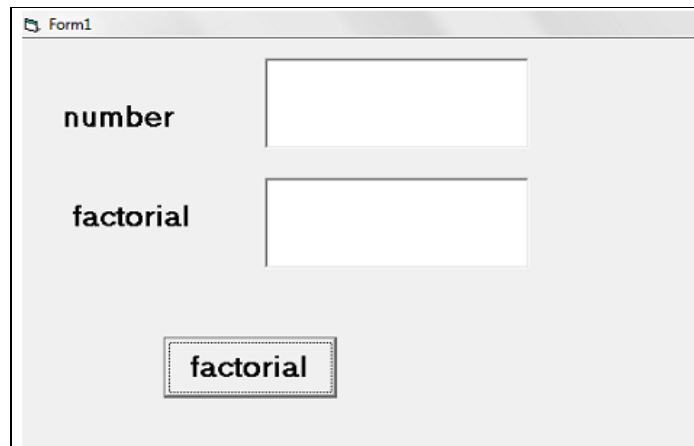
من الواضح أن الأرقام الناتجة أقل من 1 ، ويمكننا الحصول على أرقام أكبر عشوائية أكبر من 1 بضرب العدد في 10 أو 100 الخ ، ولكن سيظل لدينا جزء كسري ، ويمكننا التخلص منه بدالة العدد الصحيح INT ، الآن نكتب السطر التالي :

$z = \text{int}(10 * \text{RND}(1))$

هنا نلاحظ أن الأعداد الناتجة هي أعداد صحيحة ليست أكبر من 10.

7-3 مجموعة امثلة

مثال : اكتب برنامج لحساب مفاكوك اي عدد.



Private Sub Command1_Click()

x = Val(Text1.Text)

```
If x = 0 Then f = 0 Else f = 1
```

```
For i = 1 To x
```

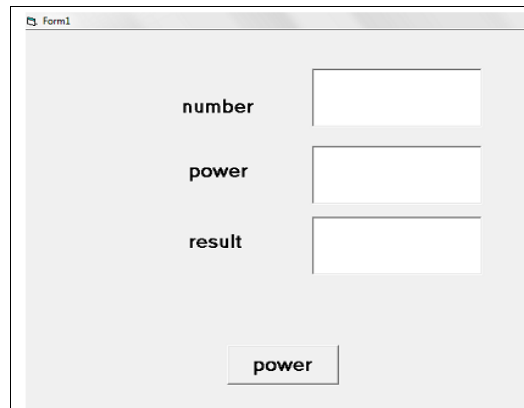
```
f = f * i
```

```
Next
```

```
Text2 = f
```

```
End Sub
```

مثال : اكتب برنامج لحساب ناتج رفع اي عدد لاس معين.

A screenshot of a Windows application window titled "Form1". The window has a light gray background. On the left side, there are three text boxes stacked vertically, each with a label to its left: "number", "power", and "result". To the right of each label is an empty text box. Below these text boxes, centered horizontally, is a button with the text "power" on it.

```
Private Sub Command1_Click()
```

```
x = Val(Text1.Text)
```

```
y = Val(Text2.Text)
```

```
For i = 1 To y
```

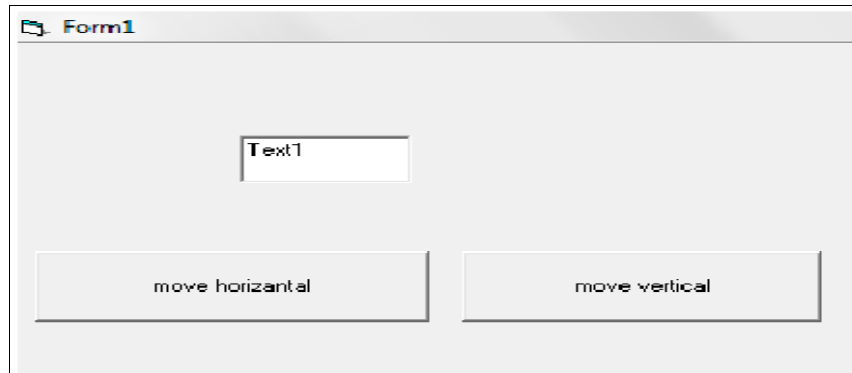
```
f = x ^ y
```

```
Next i
```

```
Text3 = f
```

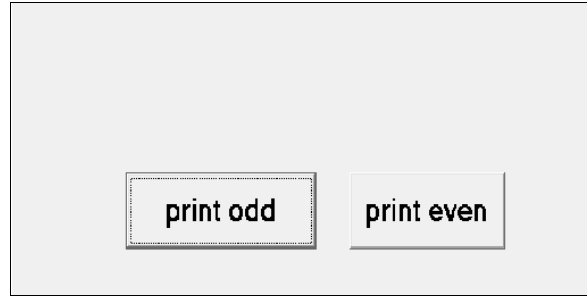
```
End Sub
```

مثال : اكتب برنامج لتحريك text افقيا و عموديا



```
Private Sub Command1_Click()  
For i = 1 To Form1.Width  
For j = 1 To 50  
Text1.Left = i Mod Form1.Width  
Form1.BackColor = QBColor((i Mod 15) Mod 2)  
Next j, i  
Text1.Left = Rnd(1) * Form1.Width  
End Sub  
Private Sub Command2_Click()  
For i = 1 To Form1.Height  
For j = 1 To 50  
Text1.Top = i Mod Form1.Height  
Form1.BackColor = QBColor((i Mod 15) Mod 2)  
Next j, i  
Text1.Top = Rnd(1) * Form1.Height  
End Sub
```

مثال : اكتب برنامج لطباعة الاعداد الفردية او الزوجية من (1-100)



```
Private Sub Command1_Click()  
Form1.Cls  
Form1.FontSize = 14  
For i = 1 To 100  
Form1.ForeColor = QBColor(i Mod (16))  
If i = 50 Then Print  
If i Mod 2 = 1 Then Print i & " ";  
Next i  
End Sub
```

```
Private Sub Command2_Click()  
Form1.Cls  
Form1.FontSize = 14  
For i = 1 To 100  
If i = 50 Then Print  
If i Mod 2 = 0 Then Print i & " ";  
Next i  
End Sub
```

مثال : اكتب برنامج لتحويل عدد من النظام العشري للثنائي

The image shows a screenshot of a Windows form titled "Form1". The form has a light gray background. On the left side, there are two labels: "decimal number" and "binary number". To the right of "decimal number" is a text box. To the right of "binary number" is another text box. At the bottom center of the form is a button labeled "convert".

```
Private Sub Command1_Click()
```

```
Text2 = ""
```

```
Dim a(100)
```

```
x = Val(Text1.Text)
```

```
For i = 1 To 100
```

```
a(i) = x Mod 2
```

```
x = x \ 2
```

```
If x = 0 Then Exit For
```

```
Next i
```

```
For j = i To 1 Step -1
```

```
Text2 = Text2.Text & a(j)
```

```
Next j
```

```
End Sub
```

3-8 المصفوفات

عبارة عن متغيرات مفهرسة تحتوي على بيانات عديدة من نفس النوع . كل مصفوفة لها اسم واحد يمكن استخدامه للرجوع الى أي عنصر فيها وذلك باقتران هذا الاسم بدليل يمثل مكان العنصر في المصفوفة , ويمكن انشاء مصفوفة لاحتواء أي نوع من انواع البيانات مثل : النصوص والاعداد الحقيقية والصحيحة وغيرها , فأنواع البيانات المتوفرة في الفيچوال بيسك هي :

Integer , Long , Single , Double , Currency , Data , String, Variant Boolean , Byte , , User-defined

واستخدام المصفوفات في البرمجة يساعد برامج قصيرة وبسيطة ذات قوة اكبر لانه يمكن بناء Loops تتعامل بكفاءة مع المصفوفات مهما كان عدد عناصرها.

الخصائص الاساسية للمصفوفة في الفيچوال بيسك:

1. اسم المصفوفة يمثل عنوان في الذاكرة , ولا يمكن تغييره اثناء تنفيذ البرنامج.
2. يمكن الاعلان عن مصفوفة لاي نوع من انواع البيانات.
3. كل وحدة بيانات منفردة في المصفوفة تسمى عنصر.
4. جميع العناصر تكون من نفس النوع.
5. جميع العناصر تكون مخزونة على التتابع في ذاكرة الحاسوب ودليل اول عنصر هو الصفر (ويمكن جعله (1 باستخدام جملة Option Base .
6. يحدد موقع العنصر بواسطة رقم الصف ورقم العمود الذي يقع فيه.
7. لكل مصفوفة حد اعلى وحد ادنى , وعناصر المصفوفة تكون محصورة بين هذين الحدين.
8. من الممكن ان تكون المصفوفة ذات بعد واحد او متعددة الابعاد.

3-8-1 هناك نوعين من المصفوفات

• المصفوفة ذات الحجم الثابت (Fixed- size array) : وهي التي يظل حجمها ثابت اثناء تشغيل البرنامج.

• المصفوفة ذات الحجم المتغير (Dynammic –size array) : وهي التي يمكن تغيير حجمها اثناء تشغيل البرنامج.

تعريف مصفوفات الحجم الثابت Fixed – size array :

هناك ثلاث طرق للاعلان عن هذا النوع على النطاق الذي سوف تغطية المصفوفة : حيث للاعلان

عن مصفوفة عامة تستخدم كلمة Public وللإعلان عن مصفوفة خاصة تستخدم كلمة Dim

Private. او تحديد الحد الاعلى والادنى للمصفوفة:

• عند الاعلان عن مصفوفة , يكتب الحد الاعلى بعد الاسم وبين الاقواس.

• لا يمكن ان يزيد الحد الاعلى عن نطاق نوع المتغير.

• الحد الادنى الافتراضي هو الصفر.

• لتحديد الحد الادنى , ينبغي كتابته باستخدام كلمة ,

مثال: اعلان عن مصفوفة Student

Dim Student (1000) As integer

وبذلك يكون عدد عناصر هذه المصفوفة 1001 عنصر . ولجعلها مصفوفة عامة , نستبدل Dim

ب Public وكالتالي:

Public Student (1 To 1000) As Integer

مثال :

الاعلان عن مصفوفة بها 51 عنصر مرقمة من 0 الى 50 من دون تحديد نوع بيانات عناصرها سيعتبر النوع

Variants مباشرة.

Dim Day Array (50)

مثال :

الاعلان عن متغير BirthDate على انه مصفوفة من التواريخ المفهرسة من 1 حتى 10

Dim BirthDate (1To10) As Date

2-8-3 المصفوفة متعددة الابعاد Multi- Dimensional Arrays

يقصد بكلمة بعد هو عدد الدلائل المستخدمة للرجوع الى عنصر في المصفوفة . تستخدم المصفوفات متعددة الابعاد لتخزين بيانات مرتبطة ببعضها البعض. مثال على ذلك. تخزين كل نقطة Pixel موجودة على شاشة الحاسوب نحتاج لتخزين احداثياتها x & y وهكذا ممكن باستخدام مصفوفة 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 ذات بعد او ثلاثة ابعاد كما ستوضح الامثلة الاتية:

-Static A(3,4) As Double

-Static A (1 To 10 < 1 To 10) As Integer

-Dim B (3,1 To 10 , 1 To 15)

ملاحظة:

- عدد عناصر المصفوفة متعددة الابعاد يمكن معرفته بضرب قيم الابعاد ببعضها
- ينبغي الاحتراس وتقادي استخدام النوع Varinat قدر الامكان لما يتطلبه من مساحة تخزينية كبيرة.

استخدام ال Loops لمعالجة المصفوفات:

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

مثال:

الجملة التالية سوف تعطي قيمة لكل عنصر من عناصر المصفوفة A وهذه القيمة تكون مرتبطة بموقع العنصر داخل المصفوفة كما سنرى:


```

Dim I As 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*10*J
Next J
Next I

```

المصفوفات متغيرة الحجم Dynamic Array :

في بعض الاحيان , لانعرف مسبقا حجم المصفوفة التي سنستخدمها في البرنامج بالضبط , وقد نريد تغيير حجم المصفوفة اثناء تشغيل البرنامج , هنا سنحتاج الى المصفوفات ذات الحجم المتغير Dynamic حيث يمكننا تغيير حجمها في وقت لاحق. تعتبر المصفوفات متغيرة الحجم احد مميزات الفيجوال بيسك , وهي تساعد في تنظيم الذاكرة بكفاءة فمثلا, يمكن استخدام مصفوفة كبيرة لوقت قصير ثم اعادة تحجيمها لتحرير مساحة من الذاكرة عندما لا نحتاجها . وها من شأنه تسريع المعالجة.

ولصناعة Dynamic Array التالي:

1. نعلن عنها بأحد أوامر الاعلان الموضحة سابقا (Public or Private or Dim or Static)

ونجعلها ديناميكية بعدم وضع أي رقم في الاقواس كما يوضح المثال التالي:

Dim Ann ()

2. نعيد الاعلان عنها مع تحديد عدد العناصر باستخدام جملة ReDim كما في المثال التالي:

ReDim Ann (x)

جملة ReDim:

تستخدم في مستوى ال Procedure لاعادة تخصيص مساحة تخزينية لمصفوفة متغيرة الحجم
. Dynamic Array

• تستخدم جملة ReDim لتجميع او اعادة تجميع مصفوفة متغيرة الحجم Dynamic

Array والتي قد اعلن عنها مسبقا باستخدام أي من الجمل Dim , Private, Public مع الاقواس الفارغة (أي بدون ذكر الابعاد).

• يمكن تكرار استخدام جملة ReDim لتغيير عدد العناصر والابعاد لمصفوفة , ومع لك لا يمكن الاعلان عن مصفوفة بنوع معين من البيانات ثم اعادة تعريفها لاحقا لنوع اخر من البيانات.

• اذا كان للمصفوفة بعد واحد فيمكن اعادة تجميع هذا البعد لانه البعد الاخير والوحيد بالمصفوفة.

• واذا كان للمصفوفة بعدين او اكثر فيمكن فقط تغيير حجم البعد الاخير مع الاحتفاظ بمحتويات المصفوفة

• اذا صنعنا مصفوفة اصغر مما كانت فان بيانات العناصر المخزنة سوف تفقد.

• لاعادة تعريف المصفوفة مع خزن البيانات كلمة Preserve في اعادة التعريف.

مثال :

لصناعة مصفوفة M كمصفوفة متغيرة الحجم , نعلن عنها اولاً على النحو التالي:

Dim M () As Integer

ثم نعيد اعلان عنها داخل ال Procedure على النحو التالي:

ReDim M(9,15)

مثال :

يبين المثال التالي كيف يمكن زيادة حجم البعد للاخير للمصفوفة متغيرة الحجم بدون محو البيانات الموجودة فيها:

ReDim Preserve X(10,10,15)

ReDim Preserve X(10,10,1)

مثال :

لادخال قائمة من الاعداد غير معلومة العدد مسبقا (أي وقت كتابة الشفرة code) ويسأل عن عددها عند تشغيل الcode :

Dim X () As Integer

("ادخل عدد العناصر من فضلك" N= InputBox)

ReDim X(N)

For I= 1 To N

("X(I) =InputBox القائمة من فضلك")

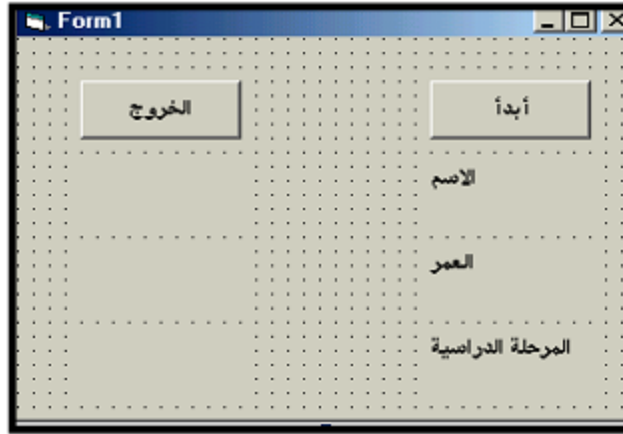
Next I

مثال :

المطلوب عمل برنامج معلومات عن الطلاب بحيث يقوم البرنامج بطلب اسم الطالب وعند ادخال اسم الطالب يعطينا البرنامج المعلومات المتوفرة عنه ؟

التنفيذ:

نصمم النافذة التالية للبرنامج حيث نضع فيها زرین احدهما يقوم بتنفيذ البرنامج Command1 , بينما الثاني للخروج من البرنامج Command2 ونضع Label عدد ستة . ثلاثة للاسم والعمر والمرحلة الدراسية والثلاثة الباقية للمعلومات المقابلة



ثم نقوم بكتابة الشفرة (code) وكالاتي:

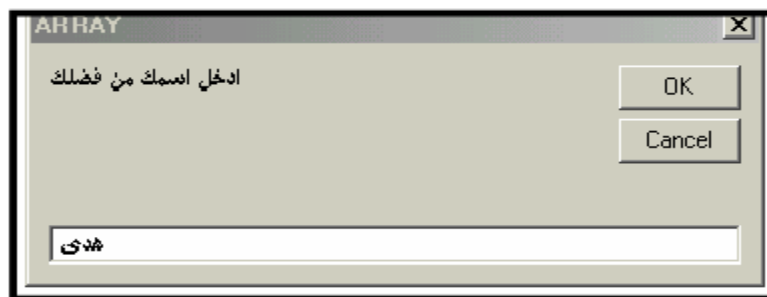
```

Project1 - المصفوفة (Code)
Command2 Click
Private Sub command1_Click()
Dim x(4) As Integer
Dim y(4) As String
Dim z(4) As String
x(1) = 34
y(1) = "فلاح"
z(1) = "الثاني"
x(2) = 23
y(2) = "عمد"
z(2) = "الثالث"
x(3) = 23
y(3) = "هدى"
z(3) = "الثاني"
x(4) = 23
y(4) = "عمر"
z(4) = "الاول"
A$ = InputBox$("ادخل اسمك من فضلك")
For I = 1 To 4 Step 1
If y(I) = A$ Then
Label14 = y(I)
Label15 = x(I)
Label16 = z(I)
End If
Next I
End Sub

Private Sub Command2_Click()
End
End Sub

```

عند تشغيل البرنامج سيطلب الاسم من خلال InputBox وعند ادخال احد الاسماء المعروفة في



لبرنامج مثلاً (هدى) وبعدها ضغط زر ok:

سيعطينا كل المعلومات المتوفرة عنها وكالتالي:

الخروج	أبدأ
هدى	الاسم
23	العمر
الثاني	المرحلة الدراسية

مثال طباعة مصفوفة ذات بعدين

```
:Dim x(2,3) as integer  
Private sub command1_click()  
For i=0 to 2  
For j=0 to 3  
Print x(i,j);  
Next  
Print  
Next  
End sub
```

مثال : مصفوفة ذات بعدين نجمع فيها الصف مع العمود:

```

Dim x(4,4) as integer
Private sub command1_click()
For i=0 to 4
For j=0 to 4
X(i,j)= i+j
Print x(i,j);
Next
Print
Next
End sub

```

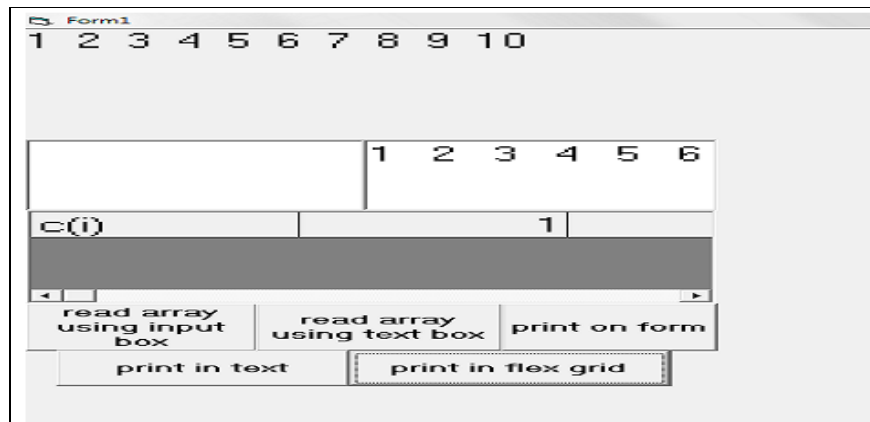
التعامل مع المصفوفات:

مثال : اكتب برنامج لادخال وطباعة مصفوفة بعد واحد

1-على ال Form

2 - داخل Text box

3- باستخدام Flex Grid



```

Dim c(10), k As Integer

```

```
Private Sub Command1_Click()  
For i = 1 To 10  
c(i) = InputBox("input element", "read array")  
Next i  
End Sub
```

```
Private Sub Command2_Click()  
k = k + 1  
c(k) = Val(Text1.Text)  
Text1.Text = ""  
Text1.SetFocus  
End Sub
```

```
Private Sub Command3_Click()  
Form1.Cls  
Form1.FontSize = 22  
For i = 1 To 10  
Form1.Print c(i) & " ";  
Next i  
End Sub
```

```
Private Sub Command4_Click()  
Text2.Text = ""  
Text2.FontSize = 22  
For i = 1 To 10  
Text2.Text = Text2.Text & c(i) & " "
```

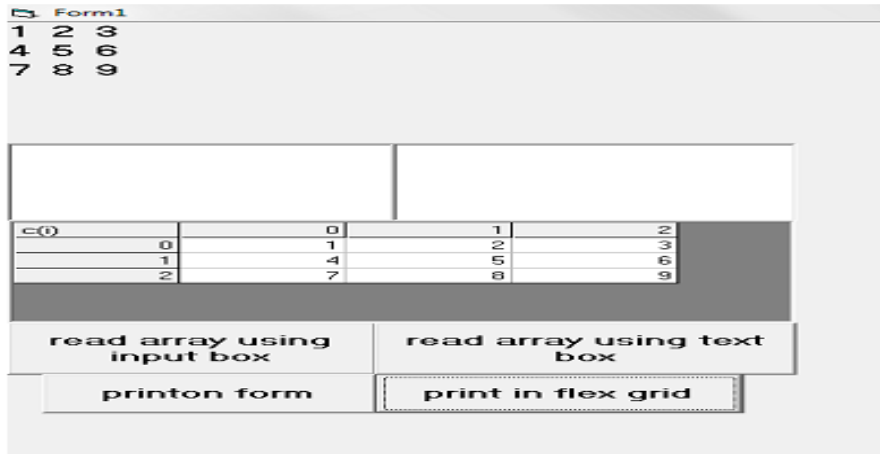
```
Next i
End Sub

Private Sub Command5_Click()
MSFlexGrid1.FontSize = 22
MSFlexGrid1.Cols = 11
MSFlexGrid1.Rows = 1
MSFlexGrid1.Col = 0
MSFlexGrid1.Row = 0
MSFlexGrid1.Text = "c(i)"
For i = 1 To 10
MSFlexGrid1.Col = i
MSFlexGrid1.Text = c(i)
Next i
End Sub
```

مثال : اكتب برنامج لادخال وطباعة مصفوفة بعددين

1-على ال-Form

2-باستخدام Flex Grid



```

Dim c(0 To 2, 0 To 2), k As Integer
Private Sub Command1_Click()
For i = 0 To 2
For j = 0 To 2
c(i, j) = InputBox("input element", "read array")
Next j, i
End Sub

```

```

Private Sub Command2_Click()
j = k Mod 3
i = (k \ 3)
c(i, j) = Val(Text1.Text)
Text1.Text = ""

```

```

Text1.SetFocus
k = k + 1
End Sub

```

```

Private Sub Command4_Click()

```

```

Form1.Cls
Form1.FontSize = 14
For i = 0 To 2
For j = 0 To 2
Form1.Print c(i, j) & " ";
Next j
Print
Next i
End Sub

Private Sub Command5_Click()
MSFlexGrid1.FontSize = 10
MSFlexGrid1.Cols = 4
MSFlexGrid1.Rows = 4
MSFlexGrid1.Row = 0
For i = 1 To 3
MSFlexGrid1.Col = i
MSFlexGrid1.Text = i - 1
Next i
MSFlexGrid1.Col = 0
For i = 1 To 3
MSFlexGrid1.Row = i
MSFlexGrid1.Text = i - 1
Next i
MSFlexGrid1.Row = 0
MSFlexGrid1.Col = 0
MSFlexGrid1.Text = "c(i)"
For i = 0 To 2

```

```

For j = 0 To 2
MSFlexGrid1.Col = j + 1
MSFlexGrid1.Row = i + 1
MSFlexGrid1.Text = c(i, j)
Next j
Next i
End Sub

```

مثال : اكتب برنامج لقلب عناصر القطر الرئيسي؟

	1	2	3			3	2	1
	4	5	6			4	5	6
	7	3	9			9	8	7
	4	5	6			1	3	2
	1	2	3			4	6	5
	7	3	9			7	9	8

```

Dim a(3, 3) As Integer
Private Sub Command1_Click()
For i = 1 To 3
For j = 1 To 3
a(i, j) = InputBox("", "")
Next j, i
table1.Cols = 4
table1.Rows = 4

```

```
For i = 1 To 3
For j = 1 To 3
table1.Col = j
table1.Row = i
table1.Text = a(i, j)
Next j, i
End Sub
```

```
Private Sub Command2_Click()
Dim b(3, 3)
For i = 1 To 3
For j = 1 To 3
b(i, j) = a(i, j)
Next j, i
For i = 1 To 3
k = b(i, i)
b(i, i) = b(i, 4 - i)
b(i, 4 - i) = k
Next i
table2.Cols = 4
table2.Rows = 4
For i = 1 To 3
For j = 1 To 3
table2.Col = j
table2.Row = i
table2.Text = b(i, j)
Next j, i
```

```

End Sub
Private Sub Command3_Click()
Dim c(3, 3)
For i = 1 To 3
For j = 1 To 3
c(i, j) = a(i, j)
Next j, i
For j = 1 To 3
k = c(1, j)
c(1, j) = c(2, j)
c(2, j) = k
Next j
table3.Cols = 4
table3.Rows = 4
For i = 1 To 3
For j = 1 To 3
table3.Col = j
table3.Row = i
table3.Text = c(i, j)
Next j, i
End Sub
Private Sub Command4_Click()
Dim d(3, 3)
For i = 1 To 3
For j = 1 To 3
d(i, j) = a(i, j)

```

```

Next j, i
For i = 1 To 3
k = d(i, 2)
d(i, 2) = d(i, 3)
d(i, 3) = k
Next i
table4.Cols = 4
table4.Rows = 4
For i = 1 To 3
For j = 1 To 3
table4.Col = j
table4.Row = i
table4.Text = d(i, j)
Next j, i
End Sub

```

مثال : اكتب برنامج لترتيب عناصر مصفوفة بعد واحد تصاعدياً؟

The screenshot shows a software interface with a grid of numbers. The grid consists of two rows and ten columns. The first row contains the numbers 2, 4, 6, 8, 0, 1, 1, 3, 5, 7. The second row contains the numbers 0, 1, 1, 2, 3, 4, 5, 6, 7, 8. Below the grid, there are two buttons: 'enter array' and 'arrang'.

2	4	6	8	0	1	1	3	5	7
0	1	1	2	3	4	5	6	7	8

enter array arrang

```

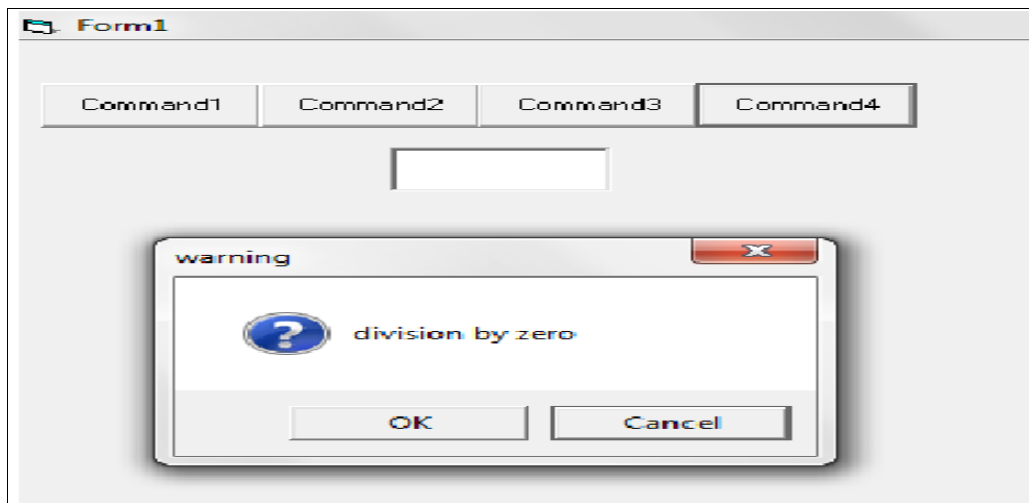
Dim a(10)
Private Sub Command1_Click()
For i = 1 To 10
a(i) = InputBox("", "")
Next i
table1.Rows = 1
table1.Cols = 11
For i = 1 To 10
table1.Col = i
table1.Text = a(i)
Next i
End Sub
Private Sub Command2_Click()
For i = 1 To 9
For j = i + 1 To 10
If a(i) > a(j) Then
k = a(i)
a(i) = a(j)
a(j) = k
End If
Next j, i
table2.Rows = 1
table2.Cols = 11
For i = 1 To 10
table2.Col = i
table2.Text = a(i)

```

Next i

End Sub

مثال : برنامج بسيط لتوضيح الرسائل التحذيرية



```
Private Sub Command1_Click()
```

```
w = MsgBox("division by zero")
```

```
Text1 = w
```

```
End Sub
```

```
Private Sub Command2_Click()
```

```
w = MsgBox("division by zero", vbOKCancel, "warning")
```

```
Text1 = w
```

```
End Sub
```



```
Private Sub Command3_Click()  
For i = 0 To 64  
w = MsgBox("division by zero", i, "warning")  
Text1 = i  
Next i  
End Sub
```

```
Private Sub Command4_Click()  
f = MsgBox("division by zero", vbQuestion + vbOKCancel +  
vbDefaultButton2, "warning")  
End Sub
```

الفصل الرابع

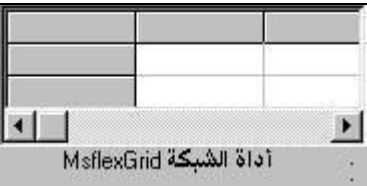

أدارة قواعد البيانات

1-4 مقدمة :

قواعد البيانات لا شك أنها تشبه جداول كبيقر للبيانات ، و بعض الوظائف دمجت بالبرنامج ادارة قواعد البيانات (مثل Access و اكسل Excel) بحيث يمكن إجراء عمليات حسابية وغيرها علي الصفوف والأعمدة ، بل والخلايا ، فضلاً عن أن برنامج اكسل Excel يمكنه تقديم وظائف أخرى مثل الرسم والتحكم في العرض .

الواقع أن أداة الشبكة يمكن تشبيهها بجدول اكسل Excel ولكن بلا وظائف ، وتوفر فيجوال بيسك لها خواص و أساليب برمجية تمكنك من التحكم في خلايا والصفوف الشبكة .
والواقع أن ميكروسوفت أنتجت أول أداة للشبكة مصاحب للإصدار الرابع من فيجوال بيسك باسم Grid ، ثم أضافت في الإصدار الخامس Msflexgrid الذي أضاف خواص أكثر للأداة ، و كذلك أضافت ميكروسوفت الأداة Mshflexgrid التي تمتعت بخواص أكثر للتعامل مع قواعد البيانات .

إدراج أداة الشبكة :

علي النموذج	في نافذة الأدوات
	

أهم خواص الشبكة :

اسم الخاصية	قيمها	وظيفتها
Cols	عدد	تحدد عدد أعمدة الشبكة
Rows	عدد	تحدد عدد صفوف الشبكة
DataSource	اسم أداة البيانات المرتبطة بقاعدة البيانات مثل الأداة Data	تظهر البيانات التي تحملها أداة قاعدة البيانات Data
AllowUserResizing	0 ، 1 ، 2 ، 3	0 عدم إتاحة 1 تغيير اتساع الصفوف أثناء التنفيذ. 2 تغيير اتساع الأعمدة أثناء التنفيذ 3 تغيير اتساع الصفوف والأعمدة أثناء التنفيذ
FixedCols	عدد	تحدد عدد الأعمدة الثابتة
FixedRows	عدد	تحدد عدد الصفوف الثابتة
ScrollBars	0 ، 2 ، 3	0 بدون شريط تمرير 1 شريط تمرير أفقي 2 شريط تمرير رأسي 3 شريطي تمرير أفقي ورأسي

كما تتمتع أداة الشبكة بنفس الخصائص السابقة مثل الظهور Visible و الخط ، و غيرها .

أهم الطرق Methods و الأوامر:

- 1- دخول الخلية EnterCell وهو حدث لدي نقر إحدى الخلايا و التعامل معها .
 - 2- حدث ترك الخلية LeaveCell و يحدث لدي ترك المستخدم للخلية .
 - 3- حدث تغيير الصفوف أو الأعمدة RowColChange و يحدث لدي تغيير عدد أعمدة الشبكة أو صفوفها من حيث الزيادة أو النقصان .
 - 4- منظومة النصوص TextArray
 - 5- و هي ترقيم الخلايا من 0 إلي آخر الخلايا فمثلا العد يبدأ من 0 إلي آخر خلية في الشبكة .
 - 6- مصفوفة أو جدول النصوص TextMatrix:
- وهو يحدد الخلية برقمها في الصف والعمود فمثلاً MsflexGrid1.TextMatrix(2,5) هي الخلية التي تقع في الصف الثاني ، والعمود الخامس منه ، حيث ان رقم أول صف وأول عمود هو 0 .
- 7- RemoveItem وهو يحذف صف معين

2-4 الأداة Data :

علي النموذج	في نافذة الأدوات
	

تستخدم أداة الشبكة في الارتباط بقاعدة بيانات سبق إعدادها ، و بالتالي فهي قناة لتوصيل البيانات لأدوات أخرى .

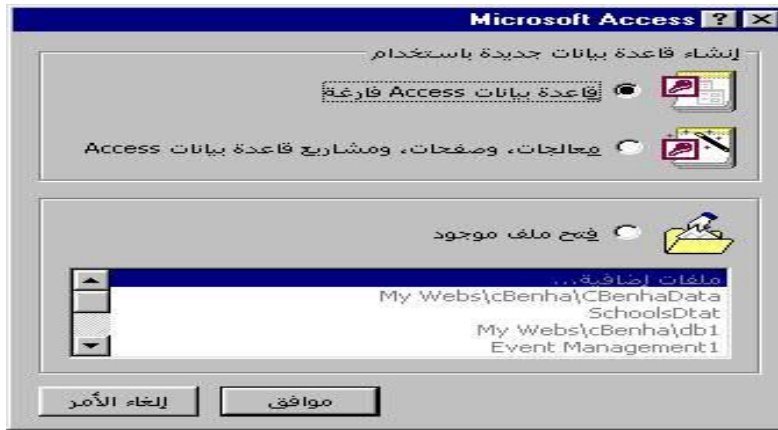
3-4 ضبط الأداة Data على قاعدة البيانات :

إنشاء قاعدة بيانات :

هناك طريقتين لإنشاء قاعدة بيانات جديدة الأولى باستخدام فيجوال بيسك، والثانية باستخدام برنامج Access وسوف نعرض طريقة إنشاء قاعدة البيانات بالطريقة الأخيرة.

4-4 طريقة برنامج أكسس :

نفتح برنامج Access حتى تظهر لك النافذة التالية :



نختار قاعدة بيانات فارغة

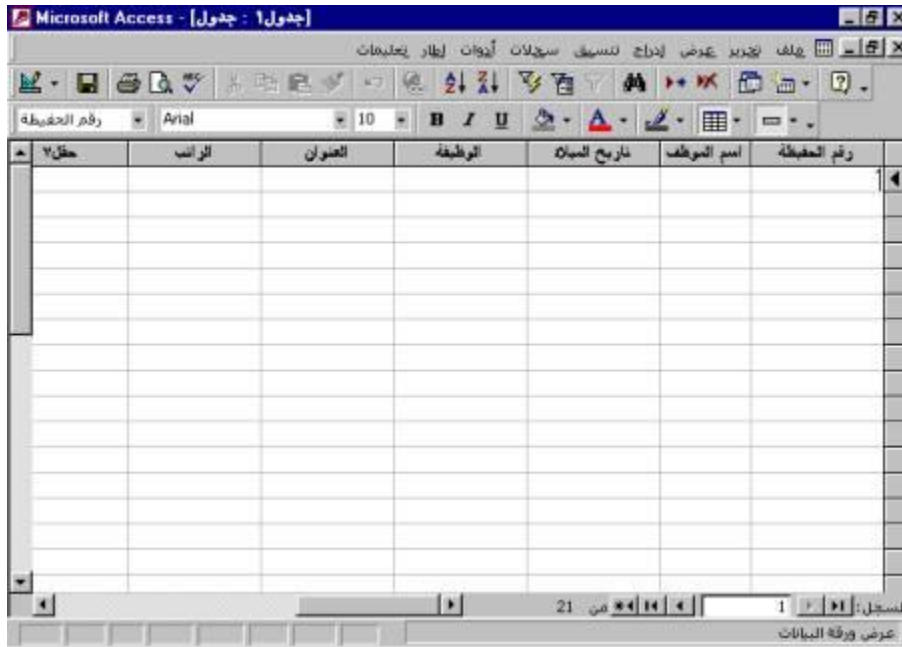


ننقر الزر إنشاء ثم تظهر لك النافذة التالية:



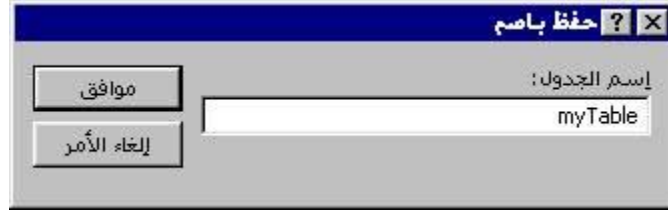
نختار جدول بإنشاء البيانات

في رأس الأعمدة - الحقول - نعيد تسميتها كما يلي :



رقم -اسم الموظف - تاريخ الميلاد - الوظيفة - العنوان - الراتب

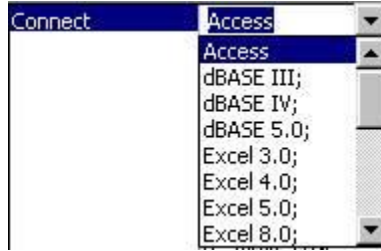
ننقر زر الحفظ فتظهر النافذة التالية :



الآن لدينا قاعدة بيانات اسمها MyDataBase و جدول اسمه MyTable

الخاصية Connect :

ننقر نافذة الخواص حتى نجد الخاصية Connect ثم ننقرها و نختار نوع قاعدة البيانات :



حيث بإمكانك ربط الأداة بأي نوع نراه بالجدول حتى Fox Pro و لكن قاعدة البيانات لها مشكلة مع Access 2000 والسبب أن فيجوال بيسك لا يدعم هذا المحرك و لكنه يدعم المحرك Jet 2 وبالتالي يجب أن نتبع الخطوات التالية لتجعل فيجوال بيسك يتعامل مع قاعدة بيانات Access 2000



ننقر قائمة أدوات

- نختار أدوات مساعدة لقاعدة البيانات .
 - نختار تحويل قاعدة بيانات
 - نختار إلى إصدار سابق من قاعدة بيانات Access .
- ثم نحفظ الملف مع ملاحظة حفظ الملف باسم جديد
لنعد للخاصية Connect ، فهي تحدد نوع قاعدة البيانات ، فمثلاً يمكننا أن نختار اكسل أو غيرها من قواعد
البيانات ، وفي حالتنا سوف نختار نوع قاعدة البيانات Access .

الخاصية DataBaseName :

هذه الخاصية تحتفظ بموضع أو عنوان قاعدة البيانات حيث لدى نقر (...) تظهر لنا نافذة فتح ننقر قاعدة البيانات وهي في حالتنا MyDataBase ثم موافق سوف نجد أن الخاصية أصبحت قيمتها : C:\My Documents\mydatabase.mdb

الخاصية RecordSetType :

و تأخذ ثلاثة قيم هم 0 و تعني جدول Table ، 1 و تعني DynaSet ، 2 و تعني SnapShot و من الضروري أن نحدد النوع هنا :

العيوب	المزايا	نوع السجل
لا ينال إلا جدولاً واحداً	- يمكنك استخدام فهرس لتحديد العرض ويظهر التحديثات التي ينفذها الآخرون فوراً	Table
لا يستخدم فهرساً ولا تظهر التعديلات التي ينفذها الآخرون فوراً	- يعمل مع جدول واحد أو عدة جداول بأكملها أو بقسم منها .	Dynaset
صعب تحديثه و يستخدم قدر كبير من الذاكرة .	سريع لأنه يعمل مع نسخة من البيانات	snapshot

و في مثالنا الحالي تكون قيمة الخاصية هو 0 أى أننا نريد جدول .

لذلك نضبط الخاصية RecordSource على جدول فلدى نقر قيمتها نجد اسم الجدول MyTable .

تلك كانت أهم الخواص و لكن كيف نعرض البيانات ؟

الواقع أن الأداة Data هي أداة لربط إحدي الأدوات بحقل من حقول قاعدة البيانات ، لو تفحصنا بعض الأدوات القياسية التي نجدها في نافذة الأدوات وهي :

أ- خانة التحقق CheckBox

ب- الخانة المركبة ComboBox

ت- الصورة Image .

ث- العنوان Label .

ج- القائمة ListBox .

ح- PictureBox .

خ- خانة النص TextBox

وهناك أيضاً أدوات يمكننا إضافتها وهي :

أ- عرض البيانات DataList .

ب- DataCombo

ت- DataGrid

ث- Microsoft Flex Grid

ج- Microsoft Hierarchical FlexGrid

ح- خانة النص الغني RichTextBox

خ- Microsoft Chart .

وحتى نربط أى من تلك الأدوات بقاعدة البيانات ، نقوم بضبط الخاصيتين التاليتين لهم :

Data1 وليكن Data	نضع اسم الأداة Data وليكن Data1
DataField	نختار منه اسم الحقل المطلوب أن تعرضه الأداة .

و لكن نضبط أولاً الخاصية DataSource على أداة ربط البيانات Data ثم ننتقل لخانه النص textbox الخاصية DataField ولدى نقرنا لقيمتها تظهر كل حقول الجدول ثم نقوم باختيار اسم الحقل الذي سوف تعرض او تعدل بياناته .

ولربط اداة الشبكة MsflexGrid نقوم بضبط الخاصية DataSource على اسم أداة ربط البيانات Data ، وأهم ما يميز أداة الشبكة هو أنها لا تحتاج اكثر من ذلك فحين نشغل البرنامج نشاهد بيانات قاعدة البيانات .ولكن لايمكننا تصحيح البيانات .

5-4 الأداة DataGrid

وهي لا تستخدم مع أداة ربط البيانات Data لأنها لا تطابقها ولكنها تستخدم أداة أخرى هي Adodc و تتميز بأنها تفتح قناة مزدوجة للتعديل و الإضافة و الحذف مباشرة مع قاعدة البيانات .

6-4 التحكم في الأداة Data

الأوامر

- تحرك للأول MoveFirst : ويعني تحرك للسجل الأول.
- تحرك للآخر MoveLast : تحرك للسجل الأخير .
- تحرك للتالي MoveNext : تحرك للسجل التالي .
- تحرك للسابق MovePrevious : تحرك للسجل السابق .
- حذف سجل Delete : حذف السجل الحالي
- تعديل السجل الحالي Update
- إضافة سجل جديد AddNew
- عدد السجلات الحالية RecordCount
- ابحث عن أول حدوث FindFirst
- ابحث عن آخر حدوث FindLast
- ابحث عن الحدوث التالي للسجل الحالي FindNext
- ابحث عن الحدوث السابق للسجل الحالي FindPrevious
- لا يوجد شئ في البحث NoMatch .

علم السجل معين Bookmark

الصيغة :

Data1.Recordset.MoveFirst

Data1.Recordset.MoveLast

Data1.Recordset.MoveNext

Data1.Recordset.MovePrevious

Data1.Recordset.Delete

Data1.Recordset.AddNew

Data1.Recordset.RecordCount

Data1.Recordset.FindFirst

Data1.Recordset.FindLast

Data1.Recordset.FindPrevious

وهناك مجموعه من Functions و التي ترجع قيم تنفيذ في البرمجة وتفحص حالات معينه :
الإقتران EOF وهو يعنى فحص قيمه ال End of Table أى إذا كان مؤشر السجلات يقف الآن على نهاية
الجدول .. وهذا ال Function يرجع قيم True أو False وتكون القيمة المرجعة True إذا كنا نقف على
آخر سجل فى الجدول .. ومثلا على كيفية الإستخدام..

If Data1.Recordset.EOF Then

Procedure

End If

الاقتران RecordCount وهذا الاقتران يقوم بإرجاع قيمه تمثل عدد السجلات في الجدول .. ويمكن
الاستفادة من هذه الدالة في عرض عدد السجلات أو في أمور أخرى.

ومثال على استخدامه ..

```
Dim NumberOrRecord As Integer
```

```
Data1.Recordset.RecordCoun
```

الدالة `AbsolutePosition` وهي ترجع رقم السجل الذي نكون واقفين عليه (أي يكون المؤشر واقف عليه)
(ويبدأ الترقيم للسجلات من الصفر .. ويمكن استخدام الدالة في أمور كثيرة ومثلا على كيفية الاستخدام..

```
Dim PosLoc As Integer
```

```
PosLoc = Data1.Recordset.AbsolutePosition
```

مثال :

ننشئ نموذج للاتصال مع قاعدة بيانات `MyDataBase` التي تضم جدول البيانات `MyTable` ، والجدول
يحتوي الحقول التالية :

رقم النفوس

اسم الموظف

تاريخ الميلاد

الوظيفة

العنوان

الراتب

ثم ننشئ نموذج و نصف أدوات عنوان و أدوات نص و 3 أزرار أوامر وأداة `Data` كما يلي:

البرنامج السابق سوف يمكننا من البحث في أى حقل بنقر عنوانه

الآن ننسخ الشفرة التالية

```

Private Sub Command1_Click()
t = Text7.Text
If t = "" Then Exit Sub
prev = Data1.Recordset.Bookmark
aa = Text8.Text
t = "[" & aa & "]" like " & Chr(34) & "*" & t & "*" & Chr(34)
'Data1.Recordset.MoveFirst
Data1.Recordset.FindFirst t
If Data1.Recordset.NoMatch Then
Data1.Recordset.Bookmark = prev
MsgBox("لم أجد ما تبحث عنه")
End If
End Sub

```

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

```
Private Sub Command3_Click()  
Data1.Recordset.Delete  
End Sub
```

```
Private Sub Label1_Click()  
Text8.Text = Label1.Caption  
End Sub
```

```
Private Sub Label2_Click()  
Text8.Text = Label2.Caption  
End Sub
```

```
Private Sub Label3_Click()  
Text8.Text = Label3.Caption  
End Sub
```

```
Private Sub Label4_Click()  
Text8.Text = Label4.Caption  
End Sub
```

```
Private Sub Label5_Click()  
Text8.Text = Label5.Caption  
End Sub
```

```

Private Sub Label6_Click()
Text8.Text = Label6.Caption
End Sub

```

مثال: التعامل مع قواعد البيانات

1-انشاء جدول من vb6

2-التعرف على اهم الايعازات و تطبيقها

التسلسل	الاسم	الدرجة الاولى	الدرجة الثانية	الدرجة الثالثة	المعدل
1	ali	90	90	100	100

Buttons: update, add, edit, delete, first, last, previous, next

Data1 grid: 1, any one, find next

```

Dim p
Private Sub Command1_Click()
Data1.Recordset.AddNew
End Sub
Private Sub Command11_Click()
Data1.Recordset.Update
End Sub
Private Sub Command2_Click()
Data1.Recordset.Edit
End Sub

```

```
Private Sub Command3_Click()
If Data1.Recordset.EOF = False Then
Data1.Recordset.Delete
Else
MsgBox "error"
End If
End Sub

Private Sub Command4_Click()
Data1.Recordset.MoveFirst
End Sub

Private Sub Command5_Click()
Data1.Recordset.MoveLast
End Sub

Private Sub Command6_Click()
'On Error Resume Next
Data1.Recordset.MovePrevious

If Data1.Recordset.BOF Then
Data1.Recordset.MoveNext
End If
End Sub

Private Sub Command7_Click()
Data1.Recordset.MoveNext
If Data1.Recordset.EOF Then
Data1.Recordset.MoveLast
End If
End Sub
```



```

Private Sub Command8_Click()
Data1.Recordset.MoveFirst
If (Text7.Text - 1) < Data1.Recordset.RecordCount Then
Data1.Recordset.Move (Text7.Text - 1)
Else
MsgBox "لا يوجد سجل بهذا الرقم"
End If
End Sub

Private Sub Command9_Click()
p = p + 1
'z = Text8.Text
'z1 = "name=" & z & ""
z = Val(Text8.Text)

z1 = "t=" & z & ""
Data1.Recordset.FindNext z1
If Data1.Recordset.NoMatch = True And p = 1 Then
MsgBox "هذا الاسم غير موجود"
Data1.Recordset.MoveFirst
p = 0
End If
If Data1.Recordset.NoMatch = True And p > 1 Then
MsgBox "لا يوجد قيد اخر"

p = 0
Data1.Recordset.MoveFirst
End If
End Sub

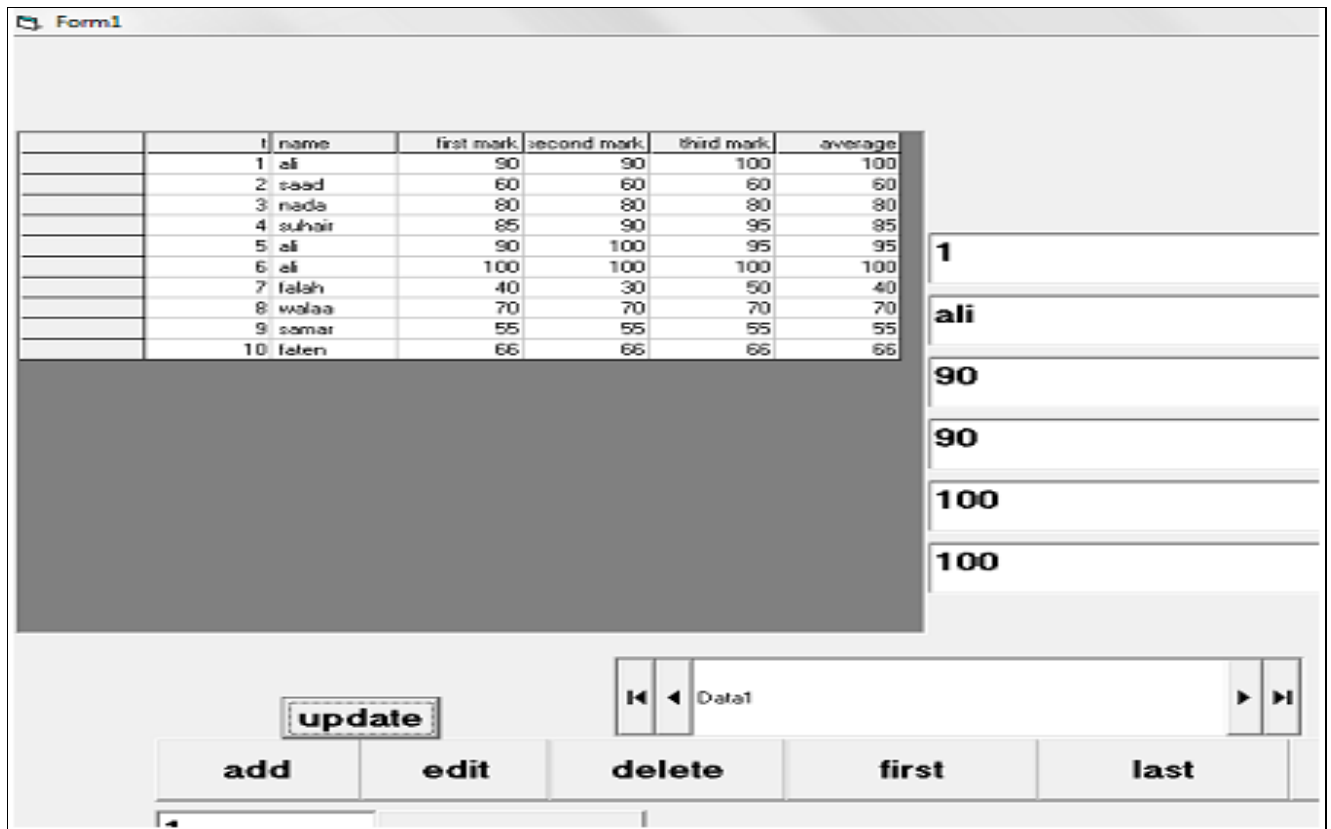
```

```

Private Sub Text8_Change()
Data1.Recordset.MoveFirst
End Sub

```

إذا اردنا عرض الجدول في flexgrid كالآتي



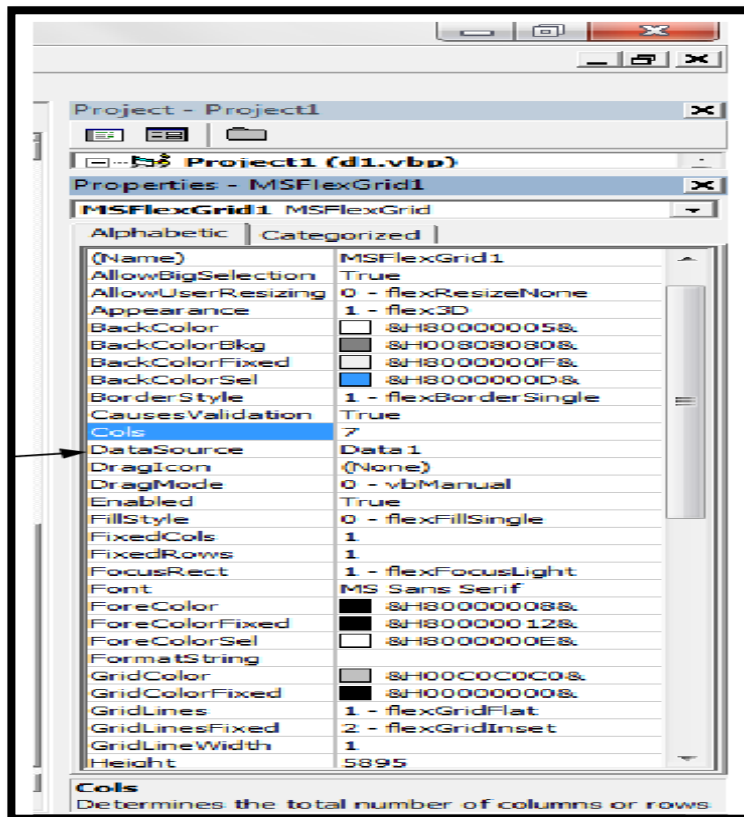
```

Private Sub Form_Load()
On Error Resume Next
MSFlexGrid1.Row = 0
MSFlexGrid1.Col = 1
MSFlexGrid1.Text = "1"
MSFlexGrid1.Col = 2
MSFlexGrid1.Text = "name"
MSFlexGrid1.Col = 3
MSFlexGrid1.Text = "first mark"

```

```
MSFlexGrid1.Col = 4
MSFlexGrid1.Text = "second mark"
MSFlexGrid1.Col = 5
MSFlexGrid1.Text = "third mark"
MSFlexGrid1.Col = 6
MSFlexGrid1.Text = "average"
End Sub
```

ويتم تعديل خصائص الـ flexgrid:



الفصل الخامس

الرسوم والصور والصوت

1-5 ادوات الرسم

هناك اسلوبان لرسم الاشكال الهندسية:

1- استخدام الدوال المكتبية مثل line & circle .

2- استخدام اداة رسم الخط (line) (واداة رسم الاشكال shape) ضمن الادوات tools .

اولا: الرسم باستخدام الدوال المكتبية (library function)

هناك اربعة دوال مكتبية تستخدم لرسم الاشكال هي:

1- الدالة line وتستخدم لرسم الخطوط والمستطيلات (الفارغة والمملوءة.)

2- الدالة circle وتستخدم لرسم الدوائر والاشكال البيضوية ومقاطع الدائرة واقواس الدائرة.

3- الدالة Pset وتستخدم لرسم النقاط.

4- الدالة print وتستخدم لرسم النصوص text كما وان هناك امران للمساعدة في السيطرة على الرسم

هي:

• الامر Scale للسيطرة على مقياس الرسم.

• الامر Drawwidth للسيطرة على سمك الخطوط.

2-5 رسم الخطوط

الصيغة العامة.

Line (x1,y1) – (x2,y2) , color

حيث ان (x1,y1) تمثل الاحداثيات السينية والصادية لبداية الخط وتقاس تلقائيا بمقياس

Twib حيث ان التويب الواحد يساوي (1\20) من البكسل Pixel (او 1\1440) من الانج الواحد.

وتمثل (x2,y2) الاحداثيات السينية والصادية لنهاية الخط بينما يمثل color اللون المطلوب للخط وحسب

جدول الالوان وفي حالة عدم ذكر اللون فان اللون التلقائي هو الاسود

سنقوم الان بعمل برنامج رسم بسيط, ليس به غير قلم فقط, الفكرة بسيطة و سهلة و الكود ايضا لن تحتوي
الفورم علي اي كائنات لان الفورم هي ما سنستخدمه للرسم عليها

كود الرسم سهلة فمثلا لو اردت رسم بالقلم تكتب:

```
If Button = 1 Then
```

```
Line – (X, Y), vbGreen ‘ لون الخط و يمكنك تغييره
```

```
End If
```

في اجراء MouseMove الخاص بالصورة ولكن هناك مشكلة, عند كتابة الكود هكذا ستجد ان هناك خط
يبدأ من اعلي الشاشة حتي المكان الذي تمشي فيه, و لمعالجة هذه المشكلة اتبع الاتي:

اكتب هذا الكود في Genral:

```
Dim DrNow As Integer
```

في هذا السطر عرفنا متغير من نوع Integer.

و هذا الكود في MouseDown:

```
DrawNow = -1
```

```
CurrentX = X
```

```
CurrentY = Y
```

و سيتم تعديل كود الرسم ايصيح كما يلي:

```
If DrNow And Button = 1 Then
```

```
Line – (X, Y)
```

End If

و اخيرا هذا الكود في MouseUp..

DrNow = 0

انواع الرسوم المختلفة

هذه طرق مختلفة للرسم و يمكنك استخدامها مثل استخدام القلم بنفس الطريقة:

● قلم:

Line -(X, Y), vbBlue

● قلم مع دوائر صغيرة:

Line -(x, y), vbGreen

Circle (x, y), 50

● رش الالوان:

Me.PSet (x + Rnd * 255, y + Rnd * 255), RGB(Rnd * 255, Rnd * 255, Rnd * 255)

Me.PSet (x + Rnd * 255, y + Rnd * 255), RGB(Rnd * 255, Rnd * 255, Rnd * 255)

Me.PSet (x + Rnd * 255, y + Rnd * 255), RGB(Rnd * 255, Rnd * 255, Rnd * 255)

Me.PSet (x + Rnd * 255, y + Rnd * 255), RGB(Rnd * 255, Rnd * 255, Rnd * 255)

3-5 رسم المستطيلات:

وتستخدم نفس الصيغة العامة للخطوط مع اضافة الحرف B للمستطيل الفرغ (Block) (او الحرف BF للمستطيل المملوء Block Full) (اي ان الصيغة العامة ستكون.

Line (x1,y1) – (x2,y2) , color, B (or BF).

الملاحظات:

يمكن السيطرة على سمك الخط او اطار المستطيل باستخدام الامر.

Draw width = 5

2-يمكن السيطرة على مقياس الرسم وبالتالي استبدال المقياس التويب بمقياس اخر وذلك باستخدام الامر.

Scale (xx1,yy1) – (xx2,yy2)

حيث تمثل xx1,yy1 الاحداثيات السينية والصادية للنقطة العلوية اليسرى (Upper – Left) للنموذج (Form) وتمثل xx2,yy2 الاحداثيات السينية والصادية للنقطة السفلية اليمنى (Lower -Right) للنموذج (Form) .

مثال : اكتب برنامج يقوم بتلوين الشاشة باللون الاخضر الفاتح مع وجود زر باسم Draw Linse

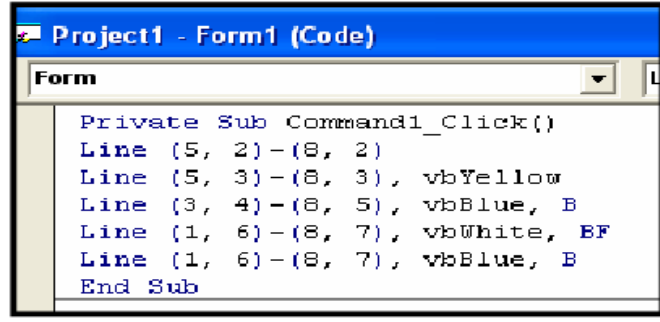
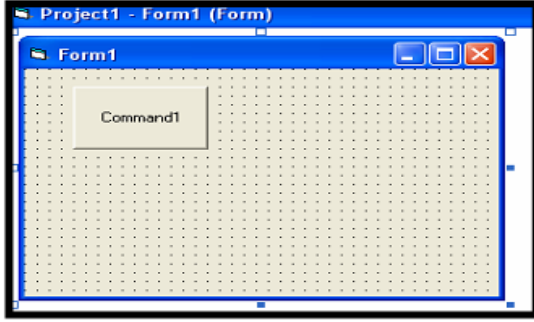
وحين الضغط على الزر باستخدام الفارة فانه سيرسم مجموعه من الخطوط والمستطيلات بالوان مختلفة.

الحل : انقر نقرة مزدوجة على اي مكان في النموذج (Form) واكتب الشفرة البرمجية التالية:

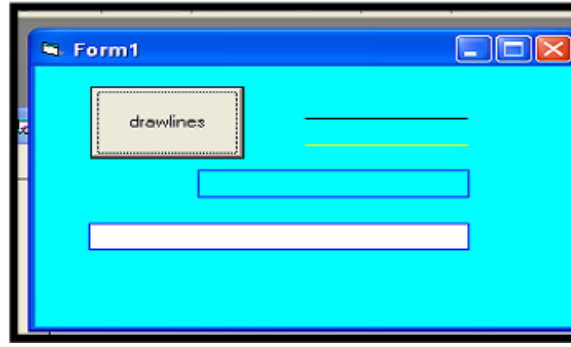
```
Private Sub Form_Load()  
Command1.Caption = "drawlines"  
Form1.BackColor = vbCyan  
Scale (0, 0)-(10, 10)  
drawwidth = 5  
End Sub
```

ضف الى النموذج اداة تحكم Command1 واضبط الخاصية التالية Draw Lines : Caption :

انقر نقرة مزدوجة على اداة التحكم واكتب الكود التالي



ونفذ البرنامج:



4-5 رسم الدوائر والاشكال البيضوية:

الصيغة العامة:

Circle (x1,y1), radius, color, start angle, end angle ,proportion

حيث ان (x1,y1) هي احداثيات مركز الدائرة و radius هو نصف قطر الدائرة و color هو لون اطار الدائرة و start angle هي زاوية بداية قوس الدائرة (مقاس بالوحدات النصف قطرية Radian) و end angle هي زاوية اية قوس الدائرة و proportion هي نسبة ارتفاع الدائرة الى عرضها (في حالة الاشكال البيضوية.)

الملاحظات:

1-في حالة ترك فراغ لموقع اللون فان اللون التلقائي هو الاسود.

2-في حالة ترك فراغ لموقع start angle, end angle فان البرنامج سيقوم برسم دائرة كاملة او شكل بيضوي كامل.

3- في حالة كتابة قيمة سالبة الى start angle فان البرنامج سيقوم برسم مقطع دائرة وليس قوس دائرة.

4- في حالة ترك فراغ لموقع proportion فان البرنامج سيقوم برسم دائرة تلقائيا اي ان default هو قيمة واحد.

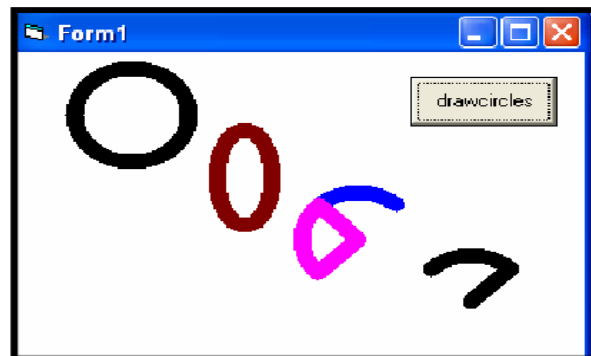
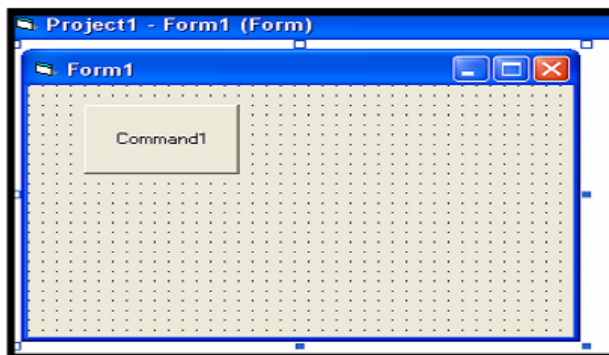
مثال : طور المثال السابق ليقوم برسم دوائر مختلفة بدلا من الخطوط.

الحل:

اكتب نفس صيغة form:

```
Command1 Click
Private Sub Command1_Click()
| draw a black circle with radius equals 1
Circle (2, 2), 1
| draw a red oval with a proportion of 2vertical to 1horizontal
Circle (4, 4), 1, QBColor(4), , , 2
Circle (6, 6), 1, vbBlue, 0.25 * 3.14, 0.75 * 3.14
'draw a sector with the same as above
Circle (8, 8), 1, vbBrown, -0.25 * 3.14, 0.75 * 3.14
'draw another sector using arc tangent function
Circle (6, 6), 1, vbMagenta, -3 * Atn(1), -5 * Atn(1)
End Sub
```

اضف الى النموذج اداة تحكم او زر امر وغير عنوانه الى draw circle .



5-5 رسم النقاط:

نقوم بتلوين نقطة على النموذج (form) (بإحداثيات x,y) (وبلون $color$ معين). الصيغة العامة لاي نقطة:

Pset (x,y) , color

مثال : طور المثال السابق ليقوم برسم ثمان نقاط واقعة على خط افقي:

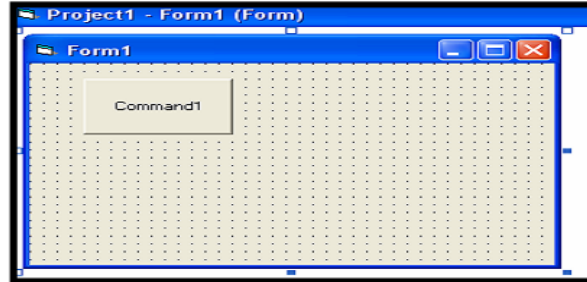
الحل:

-اكتب نفس صيغة form السابقة:

اضف الى النموذج اداة تحكم او زر امر

وغير عنوانه الى "draw".

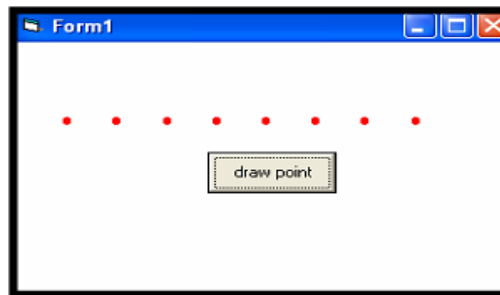
```
Private Sub Form_Load()  
Command1.Caption = "draw point"  
Form1.BackColor = vbWhite  
Scale (0, 0)-(10, 10)  
DrawWidth = 5  
End Sub
```



انقر نقرة مزدوجة على زر الامر واكتب الكود التالي:

```
Private Sub Command1_Click()  
Dim R As Single  
Dim k As Integer  
R = 0  
For k = 1 To 8  
R = R + 1  
PSet (R, 3), vbRed  
Next k  
End Sub
```

والنتائج:



6-5 الرسم باستخدام ادوات الرسم.

1-اداة رسم الخط (line) : وذلك بسحب الاداة الموجودة ضمن قائمة الادوات وتعديل

خواصها من خلال القائمة Properties .

BorderColor(to choose the color of the line) , borderstyle (soliddashed-dot-.....),
borderwidth(to choose the thickness of the line) , drawmode, visible , (true-false)
,x1,y1,x2,x2>

2- اداة رسم الاشكال (shapes) : وذلك بسحب الاداة الموجودة ضمن قائمة الادوات وتعديل
خواصها من خلال القائمة Properties .

Shape: (0-rectangle, 1-square, 2-oval, 3-circle, 4-rounded rectangle ,5-rounded square)

Fill Style : (0-solid, 1-transparent, 2-horizontal line, 3-vertical line ,4- upward diagonal, 5-downward diagonal, 6-cross, 7-diagonal cross)

7-5 اهم احداث الفورم الخاصة بالرسم

مسح كل الرسوم علي الفورم:

Me.Cls

التحكم في سمك الخط:

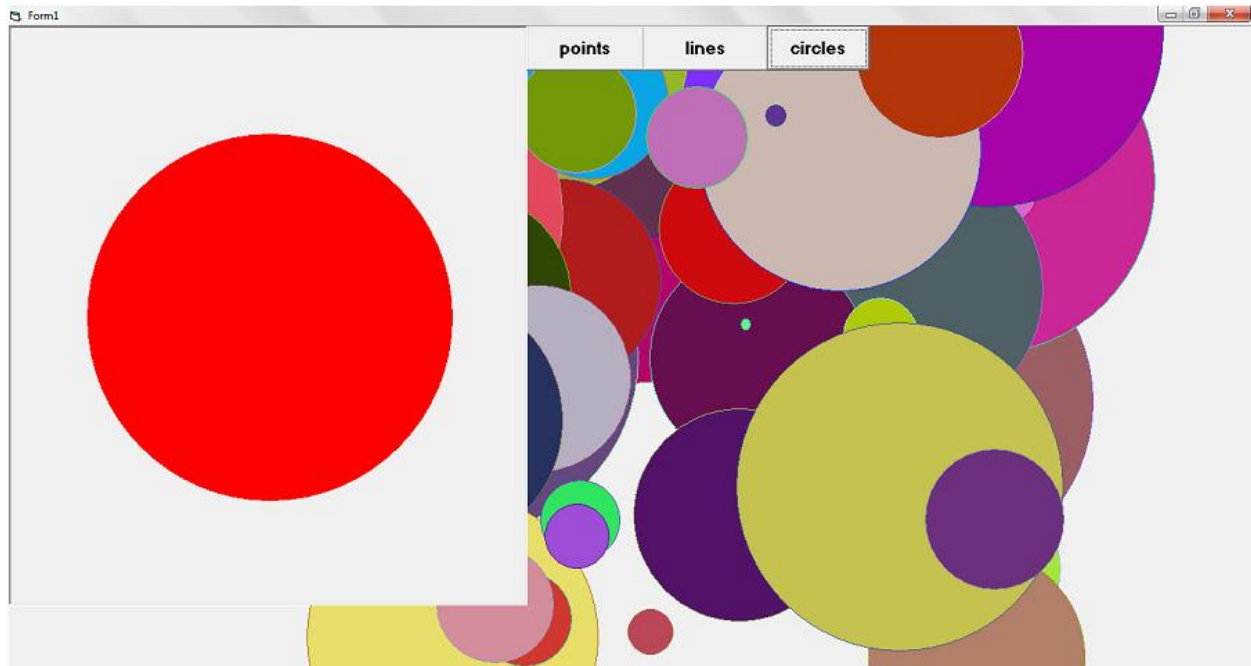
Me.DrawWidth = #

اعادة الرسوم حتي بعد مسحها:

Form1.AutoReDraw = True

8-5 امثلة

مثال : اكتب برنامج لظهار ما يلي:



```
Dim w, h, w1, h1
Private Sub Command1_Click()
For i = 1 To 100000
Picture1.PSet (Rnd() * w1, Rnd() * h1), RGB(Rnd() * 255, Rnd() * 255,
Rnd() * 255)
Form1.PSet (Rnd() * w, Rnd() * h), RGB(Rnd() * 255, Rnd() * 255,
Rnd() * 255)
Next i
End Sub

Private Sub Command2_Click()
Form1.Cls
Picture1.Cls
Form1.Line (0, 0)-(1350, 700), vbRed
```

```

Form1.Line (1100, 600)-(1350, 700), vbBlue, BF
For i = 1 To 100
Picture1.Line (Rnd() * w1, Rnd() * h1)-(Rnd() * w1, Rnd() * h1),
RGB(Rnd() * 255, Rnd() * 255, Rnd() * 255)
Next
End Sub
Private Sub Command3_Click()
Picture1.Cls
Form1.Cls
Picture1.FillStyle = 0
Picture1.FillColor = vbRed
Picture1.Circle (w1 \ 2, h1 \ 2), 200, vbRed
Form1.FillStyle = 0
For i = 1 To 50
Form1.FillColor = RGB(Rnd() * 255, Rnd() * 255, Rnd() * 255)
Form1.Circle (Rnd() * 700 + 400, Rnd() * 700), Rnd() * 200, RGB(Rnd()
* 255, Rnd() * 255, Rnd() * 255)
Next i
End Sub
Private Sub Form_Load()

w = Form1.Width
h = Form1.Height
w1 = Picture1.Width
h1 = Picture1.Height
End Sub
Private Sub Form_MouseMove(Button As Integer, Shift As Integer, X As

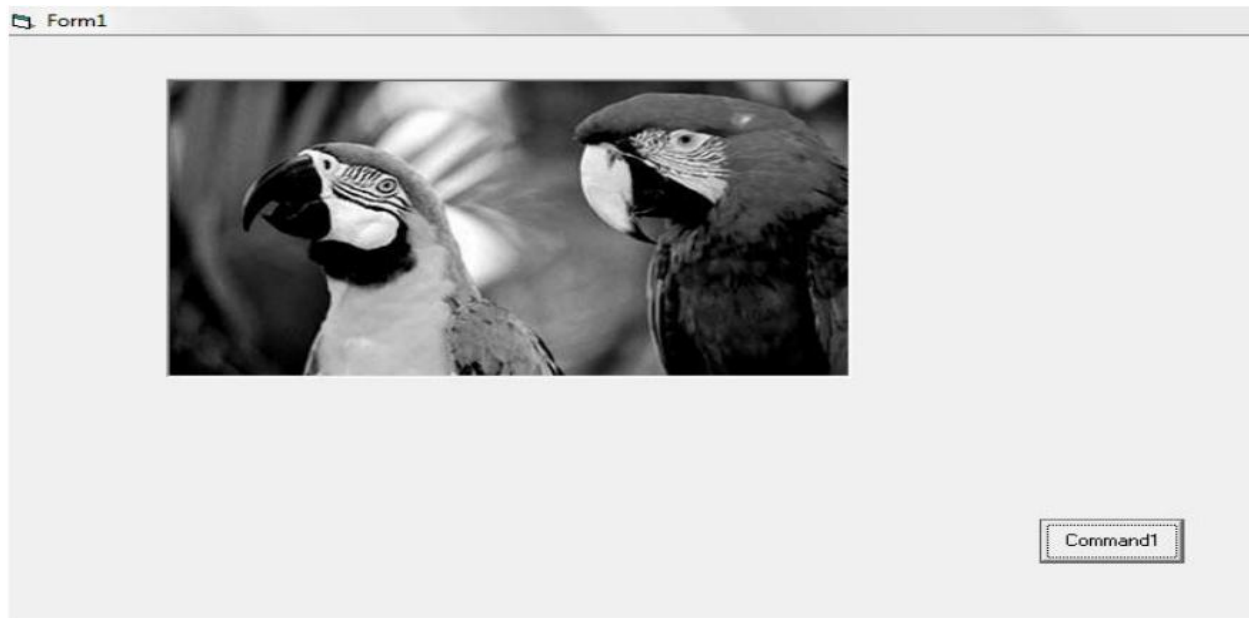
```

Single, Y As Single)

```
Form1.Line (0, 0)-(Rnd() * X, Rnd() * Y), RGB(Rnd() * 255, Rnd() *  
255, Rnd() * 255)
```

```
End Sub
```

مثال: اكتب برنامج لعرض صورة.



```
Private Sub Command1_Click()
```

```
    i = 1
```

```
    Dim temp As Byte
```

```
    Dim z1(), z22()
```

```
    Dim path As String
```

```
    Form1.CommonDialog1.DialogTitle = "Select the Bitmap Image"
```

```
    Form1.CommonDialog1.Filter = "Bmpfiles|*.bmp|"
```

```
    Form1.CommonDialog1.ShowOpen
```

```
    path = Form1.CommonDialog1.FileName
```

Open path For Binary Access Read As #1

Do

Get #1, i, temp

If i = 19 Then ww1 = temp

If i = 20 Then ww2 = temp

If i = 23 Then hh1 = temp

If i = 24 Then hh2 = temp

i = i + 1

Loop Until i = 25

Close #1

w = ww1 + (ww2 * 256) - 1

h = hh1 + (hh2 * 256) - 1

Picture1.Width = w

Picture1.Height = h

bn = (w + 1) Mod 4

i = 1

ReDim z22(((w + 1) * (h + 1) * 4) + 54)

Open path For Binary Access Read As #2

Do

Get #2, i, temp

z22(i) = temp

i = i + 1

Loop Until EOF(2)

Close #2

length = i - 1


```
k = 55
ReDim z1(w, h)

For j = 0 To h
For i = 0 To w
z1(i, h - j) = z22(k)
k = k + 3
Next i
k = k + bn
Next j

Picture1.Cls

For i = 0 To w
For j = 0 To h
Picture1.PSet (i, j), RGB(z1(i, j), z1(i, j), z1(i, j))
Next j
Picture1.Refresh
DoEvents
Next i
End Sub
```

الفصل السادس

الشبكات والانترنت

1-6 مقدمة :

دعنا نعرف بعض المصطلحات التي نستخدمها قبل أن نبدا هذا الموضوع :

1- عنوان انترنت URL :

هي اختصار للكلمات Uniform Resource Locator و تعرف أيضاً باسم عناوين صفحة الويب و هي وسيلة لمستخدمي الانترنت لإيجاد صفحات الويب ، وقد تم التصميم الأصلي في معمل الفيزياء الدقيقة الأوربي ويشار إليه بـ CERN في عام 1990م ويعبر عن طريقة لتضمين وثيقة HTML عنوان وثيقة أخرى ، و العنوان يحدد عناصر هامة للسيرفر :

- البروتوكول المستخدم .
 - السيرفر الذي يحوي الوثيقة واللوحه المفترض الاتصال بها للبروتوكول TCP و اسم المستخدم و كلمة المرور .
 - اسم الفهرس الحاوي للوثيقة مقرونا باسم الفهرس الأب .
 - اسم الوثيقة .
- والصيغة هي :

Protocol:// [username][:password][@]host[:port]/[directorypath]/[filename]

2- TCP هو اختصار للكلمات Transmission Control Protocol و تعني بروتوكول التحكم في النقل ، ويشير المصطلح TCP/IP إلي تجمع من البروتوكولات IP- تعني بروتوكول انترنت Internet Protocol - و IP يستخدم عناوين خاصة من 8 بت فمثلاً : 206.210.64.181 يحدد اسم موقع ، وما ينبغي ملاحظته هو وجود 4 أرقام تفصل بينهم ثلاثة نقاط وأقصى قيمة هي 255 ولكن نظراً لوجود صعوبة

لتذكر تلك الأرقام استخدم كلمات تسمى Top-level domain وقد نظمت الامتدادات لتشير لمؤسسات أو أهداف معينة مثلاً

government	Gov
Commercial	Com
Educational institution	Edu
Network	Net
Noncategorized organization	Org
	وهكذا

وخارج الولايات المتحدة تضاف رمز الدولة مثلا eg لمصر و sa للسعودية ، jp لليابان uk للمملكة المتحدة

ويستخدم البروتوكول TCP ما يعرف باللوحة TCP/IP لتشغيل تطبيقات محددة علي كمبيوتر معين ، وكل مقبس socket ينسب لرقم لوحة port (فيما عدا المقبسين المشتركين في لوحة) ، وعندما يطلب تطبيق موضوع علي كمبيوتر تابع أو عادي Client الاتصال بتطبيق باستخدام مقبس علي لوحة معينة ، فإن السيرفر server يبحث في التطبيقات عن المقابس المفتوحة التي تصغي listening لطلب الاتصال حتي يجد التطبيق باستخدام اللوحة port التي تم تحديدها من جانب الكمبيوتر التابع ومن ثم يمر الطلب للتطبيق ، ومعظم تطبيقات السيرفرات تحدد أرقام معينة للوحات ports لديها ، وهذا يتيح للمستعرض – فيما عدا أن اللوحة تكون مستخدمة- مثلا الاتصال علي اللوحة 80 بالنسبة HTTP :

80	HTTP(WWW)
20	FTP
21	FTP
70	Gopher
25	SMTP(e-mail)

110	POP3
23	Telnet
43	Whois
79	Finger

وعلي ذلك IP يحدد الآلة و تحدد لوحة IP تحدد التطبيق الذي يعمل علي الآلة المضيفة host - هذا يختلف عن لوحات الحاسب الذي العادي فاللوحة مجرد رقم - فهو ببساطة يحدد التطبيق .

2-6 حل شفرة الاسم Name Resolution :

وكما علمنا IP قد يكون رقم أو اسم والاسم هو الغطاء الخارجي للرقم ولا تفهمه الآلة وتوجد طريقتين لتحويل الأسماء للأرقام الرباعية السابقة القديمة وهي تخزين كل في ملف عناوين نصي يسمى HOSTS وفي نظام Unix يوجد هذا الملف في فهرس مشاركة ، وفي ويندوز 95 يوجد هذا الملف في فهرس ويندوز في كل آلة علي الشبكة وفي ويندوز الشبكات Windows NT وضع resides في Winnt\system32\drivers\etc ولهذه الطريقة عيب أظنك عرفته لأنه يتصف بالمحلية وبالتالي يحتاج للتحديث ، والطريقة المتبعة تستخدم نظام يسمى Domain Name System ويشار إليه بـ DNS وهي قاعدة بيانات توزع تحوي كل عناوين الإنترنت المسجلة .

3-6 المقبس Socket :

المقبس هو عبارة عن كل من عناوين IP و اللوحة port .
 أنتجت ميكروسوفت أداتين هامتين للتعامل مع الإنترنت هما

أ- أداة الإنترنت Inet

ب- أداة WinSocket

4-6 أداة الإنترنت Inet

لعل الترجمة الحرفية لاسم الأداة يشير لوظيفة الأداة ، فالأداة تسمى أداة نقل الإنترنت ، و تستخدم العديد من بروتوكولات الإنترنت مثل Ftp و Http و Gopher ، وتتمتع بالقدرة علي الضبط التلقائي للبروتوكول ، كما يمكنك أن الاتصال بأي موقع يعتمد تلك البروتوكولات واستدعاء retrieve الملفات باستخدام أي من الأمرين Excute أو OpenURL و هي تمكنك من :

- إضافة أي مستعرض من نوع FTP
- إنشاء تطبيق يحمل الملفات أوتوماتيكياً من مواقع FTP .
- اسناد الموقع للرسوم و تحميل الرسوم فقط .
- تقديم عرض لبيانات تفاعلية يمكن استدعائها – Retrieve – أو استردادها من الموقع .

العملية الاساسية :

تعتمد وظيفة الأداة علي البروتوكول الذي تستخدمه ، لأن البروتوكولين يختلفان بعض الشيء ، فمثلاً الأمر GetHeader يعمل فقط مع البروتوكول من النوع HTTP ، بينما توجد بعض الإجراءات المشتركة بين كل البروتوكولين الشائعين (FTP و HTTP) هم :

- ضبط الخاصية AccessType علي سرفر بروكسي صحيح Proxy Server .
- استخدام الأمر أو الطريقة OpenURL للولوج لعنوان انترنت URL صحيح .
- استخدام الأمر Execute مع عنوان انترنت صحيح وأمر يتناسب مع البروتوكول.
- استخدام الأمر GetChunk لاستدعاء بيانات الذاكرة البينية Buffer .

و إذا رغب المستخدم في الولوج إلي موقع فينبغي أن يضبط خاصية اسم المستخدم UserName و خاصية كلمة المرور Password ، ولكن العديد من المواقع تسمح بالولوج المجهول anonymous مثل موقع ميكروسوف <ftp.microsoft.com> حيث يكتب anonymous لاسم المستخدم و بريده الاليكتروني ككلمة مرور.

الطرق أو الأوامر : Methods

الوصف	الأمر أو الطريقة
<p>Inet1.OpenUrl url ,datatype</p> <p>والبارامتر datatype يأخذ قيمتين الأولى وهي الحرفية icString و الثانية الثانية icByteArray والأولى هي الفعالة وتستخدم مع البيانات النصية مثل الملفات التي امتدادها txt أو الملفات التي امتدادها htm أما ملفات الصور و الملفات التنفيذية فتحمل علي أنها icByteArray ، وفي حالة عدم استخدام url فإنه يستخدم آخر url أدخل للخاصية .</p> <p>Inet1.OpenURL("ftp.microsoft.com")</p> <p>ولكن لاحظ أنه ينبغي لفك من وعاء تضع فيه البيانات مثل صنوق نص و تصبح الصيغة الصحيحة هي :</p> <p>Text1.Text= Inet1.OpenURL("ftp.microsoft.com")</p> <p>وبالطبع تعتبر الأداة أن نوع البيانات هي بيانات حرفية لأننا لم نكتب قيمتها فتستخدم القيمة الابتدائية لها أما إذا كنت تحمل صورة فيجب أن تكتب الصيغة كما يلي :</p> <p>Pictur1.Picture= Inet1.OpenURL("ftp.microsoft.com\" pic.gif)</p> <p>ويمكنك أن تستخدم الطريقة OpenURL مع عنوان انترنت يسمح بإلحاق البيانات ، مثل عناوين الانترنت التي تسمح بالبحث القياسي ، فمثلاً الشفرة التالية تستخدم محرك بحث اسمه search.exe مع تنفيذ الشرط "find=Maui"</p> <p>strURL = "http://www.howzit.com/cgi-bin/search.exe?find=mui"</p> <p>If Inet1.StillExecuting Then</p> <p>Text1.Text = Inet1.OpenURL(strURL)</p> <p>End If</p>	<p>OpenURL</p>

<p>تستخدم الطريقة Execute مع كل من البروتوكولين FTP و HTTP</p> <p>أ- مع البروتوكول FTP</p> <p>Inet1.Execute URL ,Operation ,Data , RequestHeaders</p> <p>البارامتر URL يشير إلي عنوان الانترنت</p> <p>البارامتر operation و يشير لأحد الأوامر المطلوب تنفيذها وهي تشبه أوامر دوس وهي</p>	Execute
<p>وتستخدم لتغيير الفهرس الحالي – السيرفر – إلي فهرس آخر</p> <p>Inet1.Execute “ftp.microsoft.com” , “CD “ & “softlib”</p> <p>لاحظ ضرورة وجود فراغ بعد الأمر CD</p> <p>bussys/ deskapps/ developr/ dirmap.htm kbhelp/ misc/ peropsys/ products/ reskit/ services/ softlib/</p> <p>Inet1.Execute “ftp.microsoft.com” , “CD “ & “kbhelp”</p> <p>moneyswitcher.exe npdck.exe</p>	CD

<p>يتراجع عن الفهرس الحالي للفهرس الأب الحاوي له – علي السيرفر – و بالطبع ليس له بارامترات</p> <p>Inet1.Execute “CDUP”</p> <p>وهو يشبه أمر دوس CD..</p>	CDPU	
<p>ويستخدم لحذف ملف من علي الفهرس النشط حالياً علي السيرفر</p> <p>Inet1.Execute “DELETE “ & “MyFile.txt”</p>	DELETE	
<p>يبحث عن الملفا الموجودة في الفهرس النشط حالياً علي السيرفر – يقصد بذلك الفهرس الذي تم التوجه إليه بالأمر CD ، ويمكنك أن تحدد اسم الفهرس المطلوب استعراض محتوياته بكتابة اسم الفهرس بعد من الأمر</p> <p>Inet1.Execute “DIR “ & “mydocs”</p> <p>وهو يستعرض الملفات الموجودة بالسيرفر علي الفهرس mydocs</p>	DIR	
<p>ويستخدم في تحميل أو استيراد ملف موجود علي السيرفر وحفظه علي القرص الثابت و صيغته</p> <p>Inet1.Execute “GET file1 file2”</p> <p>وهو يقوم بتحميل الملف file1 من علي الفهرس النشط بالسيرفر إلي القرص الثابت بجهازك فمثلاً :</p> <p>Inet1.Execute “GET ServerFile LocalFile”</p> <p>وبالطبع يمكنك إنشاء إجراء يقرأ كل الملفات الموجودة بالفهرس ثم يقوم بتحميلها بجهازك عندما يفرغ بالطبع من الخطوة السابقة و إلا سوف تحصل علي رسالة خطأ تشير بأنه لا يزال ينفذ العملية السابقة .</p>	GET	
<p>يقوم بإنشاء فهرس في الفهرس النشط علي السيرفر فمثلاً :</p> <p>Inet1.Execute “MKDIR myDirName “</p> <p>وبالطبع هذا يتوقف علي نوع السماح الممنوح للمستخدم فالمستخدم المجهول لن يمكنه فعل ذلك مع السيرفرات التي تتطلب اسم مستخدم</p>	MKDIR	

وكلمة مرور		
ويستخدم في نسخ ملف Publish - Upload من القرص الثابت إلي الفهرس النشط علي السيرفر وهو يعمل عكس الأمر GET Inet1.Execute "GET LocalFile ServerFile "	PUT	
وهو يشبه الأمر DIR	LS	
يقوم بعرض أو استعراض محتويات الفهرس الحالي مع اسم الفهرس Inet1.Execute "PWD"	PWD	
يغلق الاتصال الحالي Inet1.Execute "QUIT"	QUIT	
تشبه تما الأمر GET	RECV	
ويقوم بإعادة تسمية ملف موجود فعلاً علي السيرفر بأسم جديد مثلاً: Inet1.Execute "RENAME OldFileName NewFileName" me "	RENAM E	
ويستخدم في حذف ملف من علي السيرفر وهذا يتوقف علي مدي الصلاحيه الممنوحة للمستخدم Inet1.Execute "RMDIR myDirName "	RMDIR	
وهو يشبه الأمر PUT	SEND	
يستعلم عن حجم ملف مثلاً Inet1.Execute "SIZE largefile.txy "	SIZE	
مع ملاحظة أنه إذا كان بروكسي السيرفر CERN فإن الأمر Execute لن يكون متاحاً		
يستخدم البروتوكول HTTP اربع أوامر للتعامل مع البيانات	استخدام الأمر Execute مع البروتوكول	

		HTTP
ويستخدم في استيراد أو تحميل ملف من عنوان انترنت مثلاً : Inet1.Execute "http://www.microsoft.com" & "/default.thm", "GET "	GET	
ويستخدم في استيراد عناوين الملف الموجود العنوان Inet1.Execute " HEAD"	HEAD	
يقدم بيانات إضافية تدعم الطلب من السيرفر Inet1.Execute ,POST " ,strFormData	POST	
ويستخدم في استبدال ملف بأخر Inet1.Execute ,POST " ,replace.htm"	PUT	

باقي طرق الأداة :

وتستخدم في فتح اتصال مع عنوان انترنت ويمكن أن يكون العنوان بارامتر Text1.Text = Inet1.OpenURL(strURL)	OpenURL
ويستخدم في استيراد إما كامل العنونة أو أو عنونة محددة ، فإذا لم تستخدم أى عنونة محددة فيستعيد أشياء مثل التاريخ و -Content type و Server "	GETHeader
وتستخدم في استيراد البيانات من السيرفر وتتطلب بارامترين الأول يحدد حجم البيانات المستوردة و الثاني يحدد نوع البيانات و عادة تستخدم مع StateChanged للاستدلال علي بدء عملها	GetChunk
وتستخدم لإلغاء الاتصال ووقف العملية المتداولة .	Cancel

5-6 مشروع :

أبدا مشروع جديد و ضع عليه أربع أزرار و أداة Internet وأداة قائمة وأداة نصي غني ، ونظمهم كما بالشكل التالي:



ثم قم بنسخ الشفرة التالية ووضعهما في محرر فيجوال بيسك

```
Private Sub Command1_Click()  
Dim strDirString, strFilename As String  
Dim intStart, intEnd As Integer  
Inet1.UserName = "anonymous"  
Inet1.Password = "بريدك الاليكتروني"  
strDirString = Inet1.OpenURL("http://also.as/mbadr/index.html")  
intStart = InStr(strDirString, """/")  
While (intStart <> 0)  
intEnd = InStr(intStart + 4, strDirString, """/")  
strFilename = Mid(strDirString, intStart + 2, intEnd - intStart - 2)  
If InStr(strFilename, "/") = 0 Then  
List1.AddItem strFilename  
End If  
intStart = InStr(intEnd + 4, strDirString, """/")
```

Wend

End Sub

Private Sub Command2_Click()

List1_DblClick

End Sub

Private Sub Command3_Click()

RichTextBox1.Text = Inet1.OpenURL("http://mbadr3630.bizland.com")

End Sub

Private Sub Command4_Click()

Dim bytData() As Byte

bytData() = Inet1.OpenURL("http://mbadr3630.bizland.com", icByteArray)

Open "e:\vb98\http\index.html" For Binary As #1

Put #1, , bytData()

Close #1

MsgBox"تم تنفيذ المهمة"

End Sub

Private Sub List1_DblClick()

Dim bytData() As Byte

If InStr(List1.Text, ".txt") <> 0 Then

RichTextBox1.Text = Inet1.OpenURL("ftp://http://also.as/mbadr/" &

List1.Text)

Else

```
bytData() = Inet1.OpenURL("ftp://http://also.as/mbadr/" & List1.Text,  
icByteArray)
```

```
Open "e:\vb98\ftp\" & List1.Text For Binary Access Write As #1
```

```
Put #1, , bytData()
```

```
Close #1
```

```
MsgBox"تم التحميل يا صديقي"
```

```
End If
```

```
End Sub
```

مقترحات لتطوير البرنامج :

- أضيف مربع نص يستقبل منك عنوان الإنترنت وضعه في متغير معرف حرفياً – string –
استخدم مثلاً التعريف Dim MyURL As String
- اجعل البرنامج يتعرف علي البروتوكول باستخدام الدالة InStr لتحديد ""//"" الموجودتين في كل عنوان وهي دالة تحدد رقم أول ظهور للحرفية و بالتالي اقرأ الجزء الموجود إلي يسار العنوان بالدالة left أو الدالة mid ثم حدد نوع البروتوكول .
- طور البرنامج لينشر ملف من نوع FTP (ضروري تحديد اسم المستخدم وكلمة المرور لأنك سوف تغير في الموقع يمكنك أن تفتح موقع مجاني على السيرفر www.topcites.com

6-6 الأداة WinSock :

تتيح لك الأداة WinSock الاتصال بكمبيوتر بعيد Remote أو سرفر و تبادل المعلومات معه باستخدام بروتوكول TCP أو البروتوكول UDP (User Datagram Protocol) ، وكليهما يمكنه إنشاء تطبيقات لكل من الكمبيوتر التابع client والكمبيوتر البعيد Server استخدامات الأداة :

- إنشاء تطبيقات محلية تجمع معلومات المستخدم قبل إرسالها للسرفر المركزي.
- إنشاء تطبيقات سرفر تعالج معلومات المستخدم.

- إنشاء تطبيقات المحادثة Chat .

اختيار البروتوكول :

إذا كنت تريد موافقة السيرفر علي عمليتك أو تبادل معلومات كبيرة مثل صورة ، فاختر البروتوكول FTP ، أما إذا كنت تريد متقطع intermittently أو علي مراحل أو يشعر كمبيوتر معين عندما يتم إنجاز المهام استخدم البروتوكول UDP ، فهو يشبه إرسال بريد وليس تفاعل متواصل بينما TCP لا يوجد سقف للرسالة ، حيث يتم تقسيم الرسائل الكبيرة لأجزاء و إرسالها حتي تكتمل ..

خواص الأداة :

<p>هذه الخاصية تخبرك بعدد البايتات المستقبلية حالياً في الذاكرة Buffer وهي خاصية تقرأ فقط - أي لا يمكن تغييرها - والعدد الذي يعبر عن البايتات يكون من نوع Long Integer فمثلاً :</p> <p>Myvar=MyWinsockControl.BytesReceived</p> <p>وهذا يساعدك علي معرفة المعلومات الموجودة بالذاكرة .</p>	BytesReceived
<p>هذه الخاصية تسترجع returns اسم نظام المضيف المحلي وهي للقراءة فقط وفي الظروف العادية تجد تلك الخاصية تستعيد اسم الكمبيوتر وهو الاسم الذي اخترته لجهازك عند تركيب أو تنصيب ويندوز ، ويمكنك معرفته بفتح لوحة التحكم واختيار شبكة الاتصال - تعريف .</p>	LocalHostName
<p>وهي تستعيد IP لجهازك في صيغته الرقمية وهي للقراءة فقط ونوع ما تستعيده هو حرفي .</p> <p>Myvar= MyWinsockControl.LocalIP</p>	LocalIP
<p>تستعيد و تضبط set رقم اللوحة المحلية وهي خاصية (قراءة - كتابة) ونوع ما تسترجعه Long Integer :</p> <p>Myvar= MyWinsockControl.LocalPort</p> <p>ولضبط الخاصية علي رقم اللوحة أو المنفذ port رقم 1001 نكتب :</p> <p>MyWinsockControl.LocalPort=1001</p>	LocalPort
<p>هذه الخاصية تضبط البروتوكول فلضبط الأداة علي البروتوكول FTP نكتب :</p> <p>MyWinsockcontrol.Protocol=sckTCPProtoapl</p>	Protocol

<p>ولضبطها علي UDP نكتب :</p> <p>MyWinsockControl.Protocol=sckUDPProtocol</p> <p>ولمعرفة البروتوكول المستخدم</p> <p>Myvar=MyWinsockControl.Protocol</p>		
<p>وهي تستعيد و تضبط الكمبيوتر البعيد أو عنوانه ويمكن أن تكون رقمية أو ودية – علي شكل اسم معروف – (www.microsoft.com) أو في صورة رقمية 192.143.29.47 ، مثلاً :</p> <p>Myvar= MyWinsockControl.RemoteHost</p> <p>ولضبطها علي عنوان معين :</p> <p>MyWinsockControl.RemoteHost=”www.bizland.com”</p> <p>MyWinsockControl.RemoteHost=”192.143.29.47”</p>	RemoteHost	
<p>وهو يستعيد ويضبط رقم اللوحة أو المنفذ و القيمة المستعادة من نوع long integer</p> <p>فلقراءة رقم لوحة السيرفر</p> <p>Myvar= MyWinsockControl.RemotePort</p> <p>ولضبط لوحة السيرفر أو منفذه</p> <p>MyWinsockControl.RemotePort=1001</p> <p>وبذلك يمكنك اختيار التطبيق الذي ترغب في الاتصال به علي السيرفر .</p>	RemotePort	
<p>تستعيد حالة الأداة علي هيئة عدد صحيح يعبر عن حالة الأداة:</p>	State	
<p>مغلق</p>	sckClosed	0
<p>مفتوح</p>	sckOpen	1
<p>إصغاء</p>	sckListening	2
<p>ربط الاتصال</p>	sckConnetionPending	3
<p>جاري فك شفرة المضيف</p>	sckResolvingHost	4
<p>تم فك شفرة المضيف</p>	sckHostResolved	5
<p>جاري الاتصال</p>	SckConnecting	6
<p>تم الاتصال</p>	sckConnected	7

الكمبيوتر المرافق أغلق الاتصال	sckClosing	8
يوجد خطأ	sckError	9

الطرق Method :

الطرق هي دوال معدة سلفاً تؤدي وظائف خاصة علي الأداة وسوف نناقش أهم تلك الخواص :

تستخدم تلك الخاصية مع فقط مع تطبيق السيرفر FTP ، حيث تقبل طلب الاتصال من الكمبيوتر التابع ، وحتى تستخدم تلك الخاصية يجب أن تكون الأداة في حدث الاتصال ConnectionRequest وصيغته: Private Sub MyWinsockControl ConnectionRequest(ByVal requestedID as Long) MyWinsockControl.Accept End Sub	Accept
و تستخدم لغلق الاتصال بين السيرفر والتابع : MyWinsockControl.Close	Close
وتستخدم في استعادة Retrieve بلوك البيانات الموجود حالياً في الذاكرة Buffer و اختزانه في متغير من نوع variant MyWinsockControl.GetData myvar	GetData
وتستخدم علي تطبيق السيرفر لجعله ينتظر لطلب الاتصال TCP من الكمبيوتر التابع MyWinsockControl.Listen	Listen
وتستخدم لإرسال بيانات للكمبيوتر البعيد ويستخدم في حالة السيرفر والتابع معاً . MyWinsockControl.SendData myvar	SendData

الأحداث Events :

الأحداث هي الشرارة التي تبدأ تشغيل الطرق

يقع هذا الحدث عندما يغلق الكمبيوتر البعيد الاتصال ويمكن استخدامه في نهاية الاتصال.	Close
ويقع لدي حدوث الاتصال بالكمبيوتر البعيد.	Connect
ويقع لدي يستقبل السيرفر الاتصال من الكمبيوتر التابع .	ConnectionRequest
ويقع لدي استقبال بيانات من الكمبيوتر البعيد	DataArrival

ملحوظة : في حالة الأداة WinSocket عندما يرسل جهازك بيانات يسمي سرفر وعندما يستقبل يكون تابع . أما لفظ سرفر في حالة الأداة INET فهو الكمبيوتر البعيد الذي ترسل له أو تستقبل البيانات منه.
مثال: برنامج Client و server باستخدام winsocket
لتأكد من ان المنفذ مفتوح
للتأكد من ان المنفذ مفتوح ضع اداه Winsock و استخدم هذا الشفرة:

```
Private Sub Command1_Click()
```

```
On Error GoTo opn:
```

```
Winsock1.LocalPort = Text1.Text 'هنا تكتب المنفذ المراد فحصه'
```

```
Winsock1.Listen
```

```
Text2.Text = "المنفذ غير مفتوح"
```

```
Winsock1.Close
```

```
Exit Sub
```

```
open:
```

```
If Err.Number = 10048 Then
```

```
Text2.Text = "المنفذ مفتوح"
```

```
Else
```

Text2.Text = "يوجد مشكلة"

End If

Winsock1.Close

End Sub

برنامج Client هو البرنامج الذي سيظل معك لتتحكم في جهاز الخادم Server، و لإنشاء هذا البرنامج اتبع

الاتي: الي نموذجك • Microsoft Winsock Control

ضع الكائنات كما يلي:



• اضبط اسماء الكائنات كالتالي:

Command1	- cmdConnect		- lblStatus
Command2	- cmdDisconnect		- txtIP
Command3	- cmdOpen	Text2	- txtMsg
	- cmdClose	Winsock1	- tcpClient
	- cmdMsg	Form	- frmClient

على ما اعتقد ان لا مشكلة في معرفة Command1 وغيرها فستتبين من الاسم..

كتابة الشفرة البرمجية:

اكتب هذا لشفرة في زر اتصل:

```
Private Sub cmdConnect_Click()
```

```
cmdConnect.Enabled = False ' disable the connect button
```

```
lblStatus.Caption = ".." جاري
```

```
If txtIP.Text = "" Then
```

```
" MsgBoxمن فضلك ادخل رقمIP مناسب..", vbCritical
```

```
End If
```

```
tcpClient.Connect txtIP.Text, 8686
```

```
End Sub
```

اعتقد ان بداية الشفرة واضحة تماما، اما بالنسبة لنهايتها اي في اخر سطر،
نجعل البرنامج يتصل بالسيرفر عن طريق المنفذ رقم.. ٨٦٨٦
و الان نكتب هذا الشفرة في زر قطع الاتصال:

```
Private Sub cmdDisconnect_Click()
```

```
lblStatus.Caption = "Not Connected"
```

```
cmdDisconnect.Enabled = False
```

```
cmdConnect.Enabled = True
```

```
tcpClient.Close ' اغلاق الاتصال
```

```
End Sub
```

هذه الشفرة واضحة تماما و لا تحتاج الي شرح. الان ننقر مرتين على اداه Winsock ثم اختر الاجراء
Connect و نكتب هذه الشفرة:

```
Private Sub tcpClient_Connect()
```

```
lblStatus.Caption = "Connected"
```

```
End Sub
```

و الان لنكتب الشفرة فتح و اغلاق سواقة الاقراص:

شفرة الفتح:

```
Private Sub cmdOpen_Click()  
tcpClient.SendData "opn" ' ارسال الي السيرفر  
End Sub
```

شفرة الغلق:

```
Private Sub cmdClose_Click()  
tcpClient.SendData "cls" ' ارسال الي السيرفر  
End Sub
```

و الان اخر شئ و هو زر ارسال:

```
Private Sub cmdMsg_Click()  
tcpClient.SendData "msg" & txtMsg ' ارسال الرسالة في تكست  
بوآس  
End Sub
```

الان نكون قد انتهينا من Client و الان ننتقل للسيرفر Server:

كتابة الشفرة .. Server

نفتح مشروع جديد تماما، لن نضع اي كائنات في هذا form فقط سنكتب الشفرة. نضيف اداة WinsockControl ثم نتبع الاتي:
نكتب هذه الشفرة في اجراء تحميل: Form load:

```
Private Sub Form_Load()  
tcpServer.LocalPort = 8686  
tcpServer.Listen  
End Sub
```

في هذا الكود نهئ البرنامج لانتظار ظهور رسالة من Client..
و الان نكتب هذه الشفرة في اداة Winsock في اجراء ConnectionRequest :

```

Private Sub tcpServer_ConnectionRequest(ByVal requestID As
Long)
tcpServer.Close 'close to prevent any error
tcpServer.Accept requestID 'accept all incoming requests
End Sub

```

و الآن في اجراء Error نكتب هذه الشفرة:

```

Private Sub tcpServer_Error(ByVal Number As Integer, Description
As String, ByVal Scode As Long, ByVal Source As String, ByVal
HelpFile As String, ByVal HelpContext As Long, CancelDisplay As
Boolean)
On Error Resume Next 'to prevent any more error's
tcpServer.Close 'Close the connection
tcpServer.Listen 'listen again
End Sub

```

نضيف الي مشروعنا Module و نكتب به هذه الشفرة:

```

Public Declare Function mciSendString Lib "winmm.dll" Alias
"mciSendStringA" (ByVal lpstrCommand As String, ByVal
lpstrReturnString As String, ByVal uReturnLength As Long, ByVal
hwndCallback As Long) As Long
Dim SendStr As String, ReturnStr As String

```

و نكتب هذه الشفرة في Winsock في اجراء DataArrival:

```

Private Sub tcpServer_DataArrival(ByVal bytesTotal As Long)
Dim vardata As String
Dim strdata As String
Dim cmddata As String * 3

tcpServer.GetData strdata
cmddata = Left(strdata, 3)
vardata = Right(strdata, Len(strdata) - 3)
DoCommand cmddata, vardata
End Sub

```

و اخيرا نفتح Module مرة اخرى ثم نقوم بأضافة هذه الشفرة:

```

Public Function DoCommand(command As String, data As String)
"The server is performing a command
Select Case LCase(command)
Case "opn"
SendStr = mciSendString("Set cdaudio door open", ReturnStr, 0,
0)
Case "cls"
SendStr = mciSendString("Set cdaudio door closed", ReturnStr, 0,
0)
Case "msg"
MsgBox data, vbInformation, "Information"
End Select
End Function

```

و الان ها قد انتهينا من برنامج الاتصال، يمكننا ان نضيف اليه بعض الاوامر مثل اخفاء سطح المكتب او غيرها كما يمكننا مسح بعض ملفاتها عن طريق CommonDialog.

7-6 أداة مستعرض المواقع Web Browsing

يحتاج مبرمجو فيجوال بيسك لأداتين حتى يتمكنوا من معالجة المواقع ، والربط الفائق hyperlink بتطبيقاتك

، لذا أنت بحاجة لأداتين في الوقت الحالي هما :

أ- أداة مستعرض المواقع WebBrowser .

ب-أداة مستعرض الإنترنت InternetExplore .

والواقع أن أداة مستعرض الإنترنت هي ActiveX يمكنها عرض وثائق HTML ، بينما أداة مستعرض

الإنترنت هي أداة OLE تستخدم في التحكم في مستعرض الإنترنت .

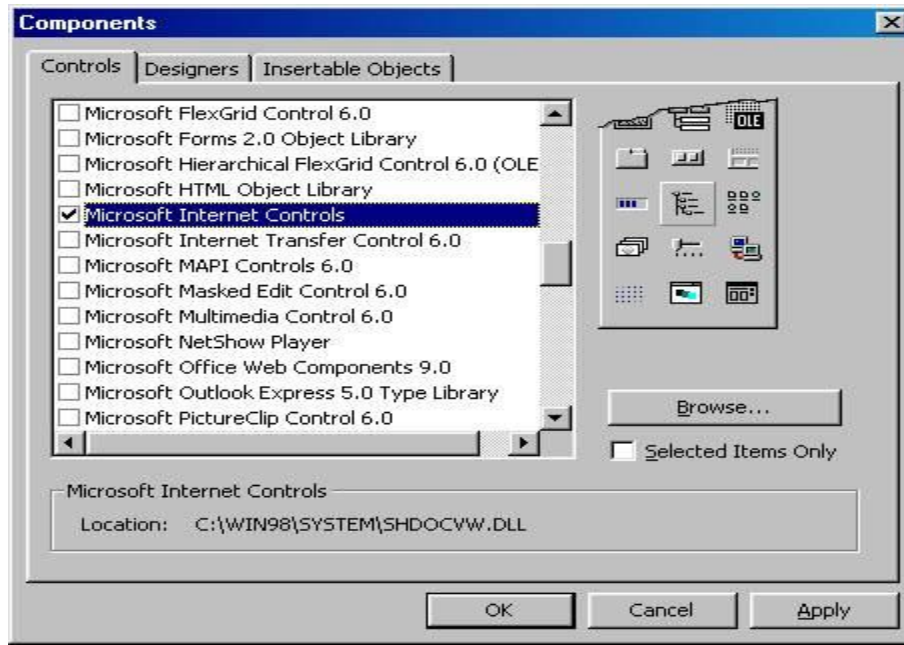
إضافة مستعرض المواقع WebBrowser :



انقر علي نافذة الأدوات بالزر الأيمن



اختر مكونات components سوف تظهر لك نافذة المكونات ، منها اختر أدوات مستعرض

الإنترنت Microsoft Internet Controls



شكل الأداة علي نافذة المشروع	شكل الأداة علي نافذة الأدوات
	

ويجب أن تحدد الحجم المناسب للأداة علي نافذة الأدوات ، لأن المستخدم لن يمكنه تغيير حجمها في وقت التشغيل ، و حتى تعرض صفحة ويب Web Page في أداة مستعرض المواقع ، استخدم خاصية الإبحار Navigate ، ويمكنك التحرك عبر قائمة URL والتي تعرض من خلال الطريقتين GoBack ، و GoForward ، فأداة مستعرض المواقع تتذكر أوتوماتيكياً قائمة عناوين المواقع التي تم زيارتها :

المشروع :

ضع الأدوات التالية :

الأداة		خواصها	
نافذة المشروع	Form1	الاسم	name
		الارتفاع	Height
		الامتداد	Width
			9400
			7200

أخضر	BackColor	الخلفية		
أداة مستعرض المواقع	Caption	العنوان		
5715	Height	الارتفاع	WebBrowser1	مستعرض مواقع الانترنت
9400	Width	الاتساع		
WebBrowser1	Name	الاسم		
CommonDialog1	Name	الاسم	CommonDialog1	أداة المربعات الحوارية
HTML فتح ملف	Caption		Command2	أداة الأوامر
Command2	Name			
URL اعرض	Caption		Command1	أداة الأوامر
Command1	Name			
إيقاف العرض	Caption		Command3	أداة الأوامر
Command3	Name			
خروج	Caption		Command4	أداة الأوامر
Command4	Name			
Combo1	Name	الاسم	Combo1	أداة القائمة المركبة
مواقع الانترنت	Text			

تعريف متغير منظومة كمصفوفة بحجم 4 وكعدد صحيح

```
Dim xxwep(4)
```

```
Dim i As Integer
```

في حدث النقر للقائمة المركبة اكتب

```
Private Sub Combo1_Click()
```

```
If Combo1.ListIndex = 0 Then xxwep(0) = "http://www.yahoo.com"
```

```
If Combo1.ListIndex = 1 Then xxwep(0) = "http://www.google.com"
```

```
If Combo1.ListIndex = 2 Then xxwep(0) = "http://www.hotmail.com"
```

```
If Combo1.ListIndex = 3 Then xxwep(0) = "http://www.facebook.com"
```

```
Label1.Caption = xxwep(0)
```

```
End Sub
```

وفي حدث النقر للزر Command1 اكتب :

```
Private Sub Command1_Click()
```

```
WebBrowser1.Navigate xxwep(0)
```

```
End Sub
```

وفي حدث النقر للزر Command2 اكتب :

```
Private Sub Command2_Click()
```

```
CommonDialog1.CancelError = True
```

```
On Error GoTo CancelOpen
```

```
CommonDialog1.Filter = "HTML Files|*.HTM|Text Files|*.TXT|All Files|*.*"
```

```
CommonDialog1.ShowOpen
```

```
If CommonDialog1.FileName <> "" Then
```

```
    WebBrowser1.Navigate CommonDialog1.FileName
```

```
End If
```

```
Exit Sub
```

```
CancelOpen:
```

```
    Exit Sub
```

End Sub

وفي حدث النقر للزر Command3 أكتب :

```
Private Sub Command3_Click()
```

```
    WebBrowser1.Stop
```

End Sub

وفي حدث النقر للزر Command4 أكتب :

```
Private Sub Command4_Click()
```

```
    End
```

End Sub

وفي حدث تحميل نافذة المشروع أكتب :

```
Private Sub Form_Load()
```

```
    Combo1.AddItem "برنامج فيجوال بيسك"
```

```
    Combo1.AddItem "اختبار"
```

```
    Combo1.AddItem "تعليم تصفح الانترنت"
```



```
    Combo1.AddItem "واستخدام المستعرضات"
```

```
    Combo1.ListIndex = 0
```

End Sub

8-6 الأداة Scriptlet

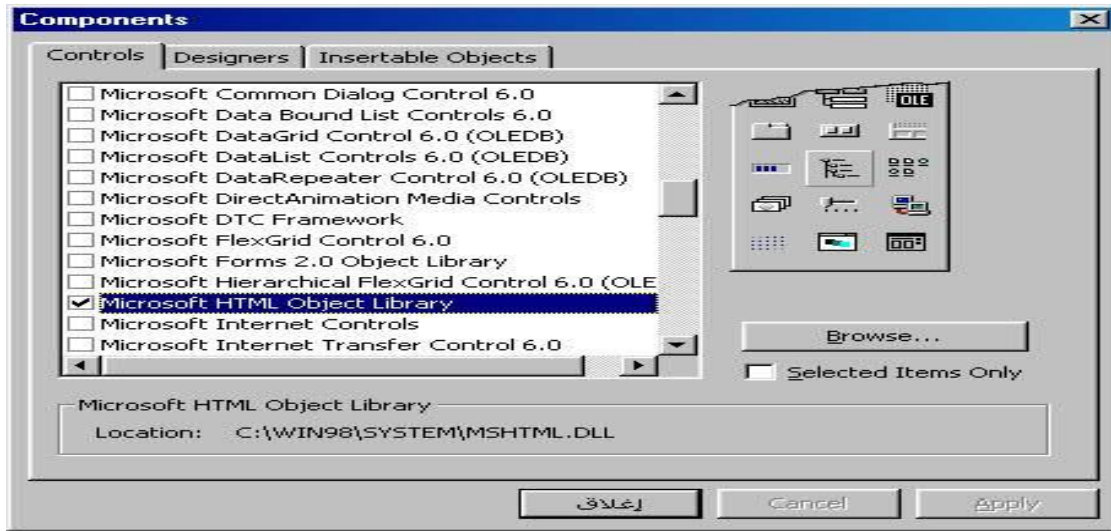
هذه الأداة تتيح لك عرض صفحة HTML – يمكن تصفحها علي مستكشف الإنترنت- ، و أبرز خواصها هي خاصية URL التي تتيح لك عرض الصفحة وسوف تشاهدها في طور التصميم ولكن لن نتمكن من التعامل معها إلا في طور التشغيل ويجب أن تضبط حجم الأداة وأن تراعي أن الأداة لا تتمتع بالخاصية Height.

في نافذة المشروع	في نافذة الأدوات
	

ويمكننا كتابة عنوان صفحتنا في الخاصية url والأداة لا تدعم تغيير العنوان أثناء التنفيذ ، وبالتالي سوف نرى فقط العنوان الذي وضعناه فيها أثناء التصميم

كيفية وضع الأداة في نافذة الأدوات :

ننقر نافذة الأدوات ونختار مكونات components ثم نعلم المربع الذي يشير للأداة وهو : Microsoft HTML Object Library



إعداد صفحة ويب Web باستخدام فيجوال بيسك

لنفرض أننا نرغب في إعداد صفحة ويب Web ، وعرضها باستخدام أدوات فيجوال بيسك ، فأنا بحاجة لعدة أمور :

-وضع أداة تدعم الحفظ بطريقة الأسكي ASCII ، وتوفر فيجوال بيسك أداتين هما أداة النص Text وأداة النص الثرية Rich Text Format والأخيرة جيدة لتعاملها مع الألوان .

-أداة مستعرض المواقع WebBrowser وشفرة توضح ذلك وربما قائمة

9-6 مشروع بناء متصفح

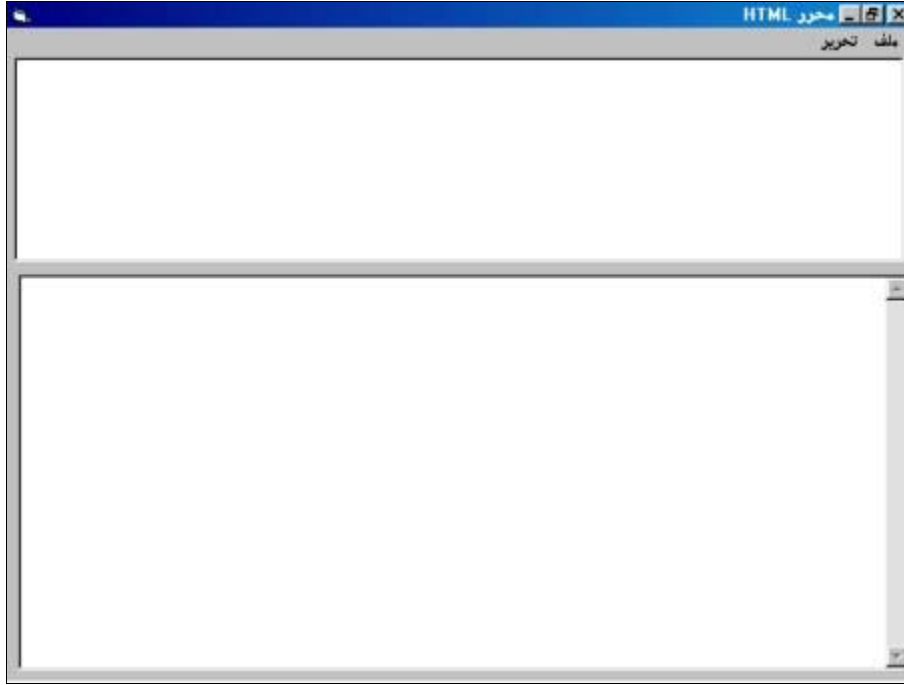
نبدأ مشروع فيجوال بيسك جديد ونضع أداة مستعرض المواقع على نافذة المشروع ، وأداة نص غنية ، وقائمة تتيح تصحيح ما أدخلناه وتحويل ما كتبناه بلغة html إلى صفحة المستعرض أ- افتح أيقونة بناء القوائم واكتب الشفرة الموجودة كما يلي :



الاسم	نوعه	العنصر
FileMenu	رئيسي-1	ملف
FileNew	فرعي	جديد
FileOpen	فرعي	فتح
FileSave	فرعي	حفظ
FileSaveAs	فرعي	حفظ باسم
NavigateTo	فرعي	فتح عنوان إنترنت
FileExit	فرعي	خروج
EditMenu	رئيسي	تحرير
EditCopy	فرعي	نسخ

EditCut	فرعي	قص
EditPaste	فرعي	لصق
EditClear	فرعي	مسح
EditAll	فرعي	اختيار الكل
ShorcutSeparator1	فرعي	-
EditRender	فرعي	ترجمة وتحويل

لا نضع أسماء في المرحلة الحالية ، ونضبط الأدوات بحيث نضع أداة النص الغنية في أعلى ، وأداة مستعرض المواقع في الجزء الاسفل من نافذة المشروع وحيث سوف تبدو نافذة المشروع كما يلي :



ضع الشفرة التالية :

Dim RestoreText

Dim OpenFileName

Dim ResizeWindows As Boolean

Dim DragStartY As Integer

Dim HTMLHeight As Integer, WebHeight As Integer

Private Sub EditAll_Click()

اختيار كل محتوى أداة النص الغنية‘

RichTextBox1.SelStart = 0 ‘بدء الاختيار من الحرف رقم صفر (الأول)‘

RichTextBox1.SelLength = Len(RichTextBox1.Text) ‘نهاية الاختيار طول حروف ‘
محتوي الأداة

End Sub

Private Sub EditClear_Click()

تفريغ أداة النص من النصوص‘

RichTextBox1.Text = ""

End Sub

Private Sub EditCopy_Click()

تنظيف محتوى الحافظة ووضع محتوى أداة النص المختار فيه‘

Clipboard.Clear

Clipboard.SetText RichTextBox1.SelText

End Sub

Private Sub EditCut_Click()

تفريغ الحافظة‘

Clipboard.Clear

نضع المحتوى المختار بالحافظة ‘

```
Clipboard.SetText RichTextBox1.SelText
```

تفريغ أداة النص من النصوص‘

```
RichTextBox1.SelText = ""
```

```
End Sub
```

```
Private Sub EditPaste_Click()
```

نضع محتوى الحافظة في المتغير Temp‘

```
Temp = Clipboard.GetText(vbCFText)
```

```
SelTextStart = RichTextBox1.SelStart بداية الاختيار‘
```

```
SelTextEnd = Len(Temp) نضع بداية الاختيار في المتغير‘
```

```
RichTextBox1.SelText = Temp نضع محتوى متغير الحافظة في الأداة‘
```

```
RichTextBox1.SelFontName = RichTextBox1.Font
```

```
RichTextBox1.SelBold = False
```

```
RichTextBox1.SelItalic = False
```

```
RichTextBox1.SelUnderline = False
```

```
RichTextBox1.SelColor = vbBlack نجعل خلفية الأداة أسود‘
```

```
End Sub
```

```
Private Sub EditRender_Click()
```

```
RenderDocument
```

```
End Sub
```

```
Private Sub EditUndo_Click()
```

```
RichTextBox1.Text = RestoreText
```

```
End Sub
```



```
Private Sub EditSelect_Click()
```

```
    RichTextBox1.SelStart = 0
```

```
    RichTextBox1.SelLength = Len(RichTextBox1.Text)
```

```
End Sub
```

```
Private Sub FileExit_Click()
```

```
End
```

```
End Sub
```

```
Private Sub FileNew_Click()
```

```
    RichTextBox1.Text = ""
```

```
    OpenFileName = ""
```

```
End Sub
```

```
Private Sub FileOpen_Click()
```

```
On Error Resume Next
```

```
    CommonDialog1.Filter = "HML Documents|*.htm;*.html|ActiveX
```

```
Documents|*.vbd|All Files|*.*"
```

```
    CommonDialog1.ShowOpen
```

```
    If Trim(CommonDialog1.FileName) = "" Then Exit Sub
```

```
    dPos = InStr(CommonDialog1.FileName, ".")
```

```
    If dPos > 0 Then ext = Mid$(CommonDialog1.FileName, dPos + 1)
```

```
    If UCase$(ext) = "HTM" Or UCase$(ext) = "HTML" Or UCase$(ext) =  
"TXT" Then
```

```
        RichTextBox1.LoadFile CommonDialog1.FileName, 1
```

```
        WebBrowser1.Navigate CommonDialog1.FileName
```

```
        OpenFileName = CommonDialog1.FileName
```

```
    End If
```

'The following lines handle non-HTML file types ' like sounds and images

```
WebBrowser1.Navigate CommonDialog1.FileName  
End Sub
```

```
Private Sub FileSave_Click()  
    If OpenFileName <> "" Then  
        RichTextBox1.SaveFile OpenFileName, 1  
    Else  
        FileSaveAs_Click  
    End If  
End Sub
```

```
Private Sub FileSaveAs_Click()  
    CommonDialog1.DefaultExt = ".htm"  
    CommonDialog1.Filter = "HTML Documents|*.htm|All Files|*.*"  
    CommonDialog1.ShowSave  
    If CommonDialog1.FileName = "" Then Exit Sub  
    RichTextBox1.SaveFile CommonDialog1.FileName, 1  
    OpenFile = CommonDialog1.FileName  
End Sub
```

```
Private Sub Form_MouseDown(Button As Integer, Shift As Integer, X As  
Single, Y As Single)  
    If Button = 1 And (Y > RichTextBox1.Top + RichTextBox1.Height) And (Y  
< WebBrowser1.Top) Then  
        Screen.MousePointer = vbSizeNS  
        ResizeWindows = True  
        DragStartY = Y
```

```

    WebHeight = WebBrowser1.Height
    HTMLHeight = RichTextBox1.Height
End If
End Sub

Private Sub Form_MouseMove(Button As Integer, Shift As Integer, X As
Single, Y As Single)
On Error Resume Next
    If ResizeWindows Then
        RichTextBox1.Height = HTMLHeight + (Y - DragStartY)
        WebBrowser1.Move WebBrowser1.Left, RichTextBox1.Top +
RichTextBox1.Height + 120, WebBrowser1.Width, WebHeight - (Y -
DragStartY)
        HTMLEdit.Refresh
    End If
End Sub

Private Sub Form_MouseUp(Button As Integer, Shift As Integer, X As Single,
Y As Single)
    ResizeWindows = False
    Screen.MousePointer = vbDefault
End Sub

Private Sub Form_Resize()
    RichTextBox1.Width = HTMLEdit.Width - RichTextBox1.Left - 200
    WebBrowser1.Width = RichTextBox1.Width
    RichTextBox1.Height = 0.3 * HTMLEdit.Height
    WebBrowser1.Move WebBrowser1.Left, RichTextBox1.Top _
+ RichTextBox1.Height + 120, WebBrowser1.Width, HTMLEdit.Height _

```

```
- RichTextBox1.Top - RichTextBox1.Height - 940
  WebBrowser1.Navigate App.Path & "\\empty.htm"
End Sub
```

```
Private Sub NavigateTo_Click()
  URL = InputBox ("أدخل عنوان الإنترنت")
  If URL <> "" Then
    WebBrowser1.Navigate URL
  End If
End Sub
```

```
Private Sub RichTextBox1_MouseDown(Button As Integer, Shift As Integer, X
As Single, Y As Single)
  If Button = 2 Then
    PopupMenu shortcutMenu
  End If
End Sub
```

```
Private Sub ShortcutAll_Click()
  Call EditAll_Click
End Sub
```

```
Private Sub ShortCutClear_Click()
  HTMLEdit.RichTextBox1.Text = ""
End Sub
```

```
Private Sub ShortcutCopy_Click()
```

```
    Call EditCopy_Click
```

```
End Sub
```

```
Private Sub ShortcutCut_Click()
```

```
    Call EditCut_Click
```

```
End Sub
```

```
Private Sub ShortcutPaste_Click()
```

```
    Call EditPaste_Click
```

```
End Sub
```

```
Private Sub ShortcutRender_Click()
```

```
    RenderDocument
```

```
End Sub
```

```
Private Sub WebBrowser1_Validate(Cancel As Boolean)
```

```
    Debug.Print"خطأ"
```

```
End Sub
```

نختبر البرنامج بما يلي :

نضع شفرة HTML التالية في النافذة العلوية :

```
<HTML>
```

```
<HEAD>
```

```
<TITLE>Document title</TITLE>
```

```
</HEAD>
```

```
<BODY>
```

```
<H1 center> من خلال فيجوال بيسك <HTML> مثال لوثيقة من نوع H1 </H1>
```

<HR>

<H3/>مكونات الوثيقة<H4>

<HR>

<H3 align="right"> < وأوامر ، وأصوات ، وصور ، ونصوص ;HTML<center>

</H4>

<HR>

<p align="right"> /إعداد الدكتور / صادق عبدالعزيز</p>

<input type=button name='button1' value="المرحلة الثالثة"=

<iframe> width=600 height=300>

</iframe>

<Script LANGUAGE="VBScript">

Document.Write "<h1/> مرحباً <h1>"

xx =inputbox("أدخل اسمك من فضلك")

yy & "...."& "مرحباً بك" =xx

msgbox yy

Document.bgcolor="#h00ff00"

Document.fgcolor="#h000cc"

sub button1_onClick ()

newurl=inputbox("من فضلك أكتب عنوان الموقع")

if newurl="" then

window.frames(0).location.href="http://www.google.com"

else

```
window.frames(0).location.href=newurl
```

```
end if
```

```
end sub
```

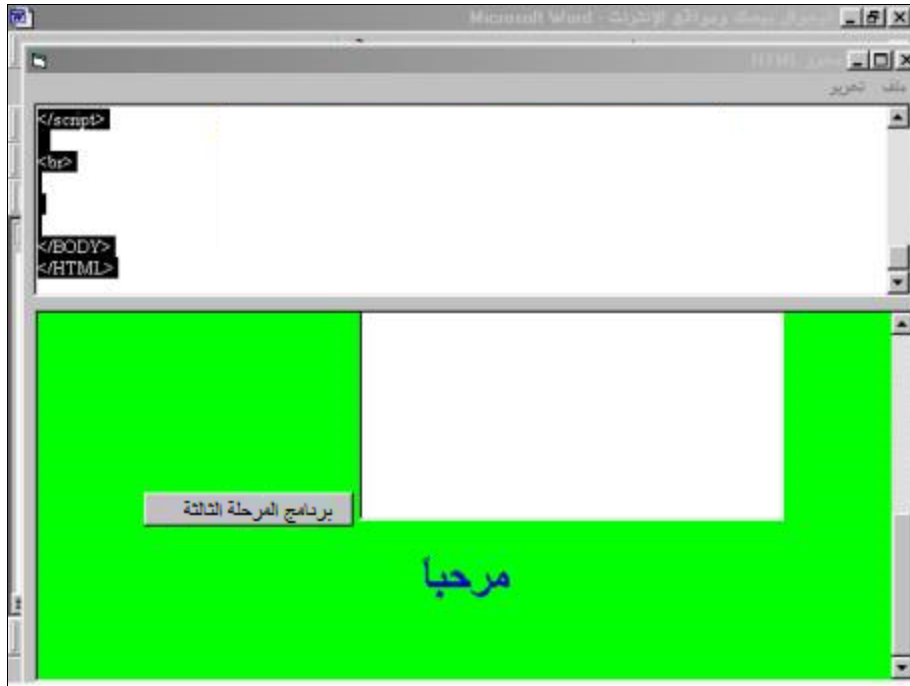
```
</script>
```

```
<br>
```

```
</BODY>
```

```
</HTML>
```

ننقر الزر الأيمن ونختار ترجمة وتحويل سوف نرى تحول تلك الشفرة إلي صفحة ويب Web في النافذة السفلى



الآن نحفظ الملف ، ونحاول استعراضه عن طريق مستكشف الإنترنت .

10-6 لغة VB Script

هي نسخة مشتقة من Visual Basic تستخدم لكتابة أود يتم ترجمته وتنفيذه بواسطة عدة برامج مثل Internet Explorer أو Microsoft Outlook..

لغة Vbscript ليس لها قوة Visual Basic ولكن قوتها ليست في اتساع اللغة ولكن في امكانية استخدامها في حالات نعجز عن استخدام اللهجات الأخرى فيها.

مثالين علي لغة VBScript

فتح أى منسق الكلمات مثل Notepad. نقوم بكتابة السطور التالية فيه

```
<HTML>
<Head>
<Title></Title>
</Head>
<Body>
<P><Script Language="VBScript">
MsgBox "مرحبا بك في كتابة برنامج الفيچوال بيسيڪ"
</Script></P>
</Body>
</HTML>
```

نقوم بحفظ الملف باسم Hllo.htm

طبعا موضح امامنا ان لتعريف لغة VBScript آتبنا الاسطر التالية:

```
<Script Language="VBScript">
</Script>
```


و الان لنقوم بعمل مثال اخر:

فتح أى منسق للكلمات مثل Notepad. نقوم بكتابة السطور التالية فيه:

```
<HTML>
<Head>
<Title></Title>
</Head>
<Body>
<INPUT TYPE=Button value=Button1 name=Button1>
<P><Script Language="VBScript">
Sub Button1_onClick()
MsgBox"2مرحبا بك في برنامج الفيچوال بيسك"
End Sub
</Script></P>
</Body>
</HTML>
```

نقوم بحفظ الملف باسم Hllo2.htm

نقوم بفتح الملف و عند الفتح سيظهر لنا زر، نضغط عليه ستظهر هذه الرسالة(مرحبا بك في برنامج الفيچوال بيسك2).

ملاحظات هامة

-لغة VBScript تدعم نمط واحد من البيانات و هو Varient..

و لا تدعم اي مصفوفات..

-ناحية مجموعات التحكم لا تدعم DoEvents ولا Goto ولا OnError ولا With ولا Numbers

Line

-من ناحية دوال التحويل لا تدعم Str و لا Val ..

-من ناحية التنقيح لا تدعم الكائن Debug ولا عبارة Stop ...

-من ناحية الاعلان عن المتغيرات لا يمكن الاعلان عن خصائص ولا يوجد تفرقة بين Public و Private وأمثالها كذلك لا يوجد حجز ديناميكي للذاكرة.

- من ناحية التحكم في الاخطاء لا يوجد دعم لـ On Error ولا Resume ولا أمثالها

- لا يوجد أى دوال للدخال أو الاخراج من ملف أو Port ..

- لا يوجد دعم للحافظة Clip board ..

- لا يوجد دعم للمتغيرات الحرفية ذات الطول الثابت وأذلك بعض دوال الحرفيات ..

انشاء الكائنات بالشفرة البرمجية

اضافة OptionButton

```
<INPUT TYPE=radio VALUE=Radio CHECKED NAME=R1>
```

اضافة ComboBox:

```
<SELECT NAME=check size=1 tabindex=1>
```

```
</SELECT>
```

انشاء نص:

```
<TEXTAREA ROWS=2 NAME=textarea COLS=20>
```

```
</TEXTAREA>
```

نص بفقرة:

```
<INPUT TYPE=text NAME=text SIZE=20 VALUE=Text><P>  
</P>
```

اضافة Button

```
<INPUT TYPE=Button value=Button name=Button1>
```

في جميع هذه الكائنات:

القيمة كما هي Name و Caption هي Value.

11-6 استخدام فيجوال بيسك إسكربت Visual Basic Script و قياسية

المستعرض Explorer

يمكننا استخدام فيجوال بيسك إسكربت Visual Basic Script أو جافا إسكربت JavaScript مع المستعرض دون الخوف من مشاكل طارئة فالمستعرض يدعمهما ، فهو يتيح لنا أن نستخدم أيهما ، فإذا قررنا ان نستخدم فيجوال بيسك إسكربت Visual Basic Script ، فنبدأ بالسطر التالي :

```
<script language="VBScript">  
"  
"  
</script>
```

و فيما يلي صيغتي المثال السابق بكائنا اللغتين .

<pre><script language="vbscript"> Sub entrance_onClick returnvalue=MsgBox (" أنت بصدد الانتقال الموقع الانتقال لصفحة البداية",52,"!تحياتي") If returnvalue=6 Then window.location="http://www.google.c om" Else window.location="http://www.google.c om" End If End Sub </script> <form> <input type="button" name="entrance" </form></pre>	<pre><script language="JavaScript"> function entrance2(){ if(navigator.appName!="Netscape") return if (confirm("You are about to enter a DHTML intensive site. Proceed?")) window.location="http://www.google.c om" else window.location="http://www.google.c om" } </script> <form> <input type="button" name="entrance" value=" google انقر هنا للدخول لموقع صفحة "> </form></pre>
--	--

value=" googleانقر هنا للدخول لموقع "> </form>	
---	--

مثال:

البرنامج التالي يطلب من إدخال رقم رسالة التنبيه و من ثم يخرجها لنا مع رقم الزر الذي نقرناه من الرسالة :

```
<script language="VBscript">
```

```
x=inputbox(0,"رسالة التنبيه","أدخل رقم صندوق التنبيه الاختياري")
```

```
w=x" هذا هو صندوق التنبيه للعدد"&
```

```
tit="صندوق رسالة التنبيه"=
```

```
retval= ( MsgBox (w ,x,tit))
```

```
y & "=retval " أنت اخترت الزر رقم="&
```

```
msgbox y ,0,"رقم زر الذي اخترته",
```

```
</script>
```



صندوق الإدخال :

في الصفحة السابقة تعرضنا لاستخدام صندوق الإدخال ، ووجدنا أن استخدامه يشبه استخدامه في فيجوال بيسك وصيغة الأمر :

InputBox(helpfile, context)[, ص][, س][, القيمة الفعالة][, العنوان][, الرسالة])



<script language="VBscript">

```
xx=InputBox("القيمة الابتدائية", "العنوان", "الرسالة", 150, 140)
```

```
</script>
```

```
<script language="VBscript">
```

```
YourName=InputBox("أدخل اسمك من فضلك", "إدخال اسم المستخدم", "Visual Basic For All")
```

```
password= inputBox("من فضلك أكتب كلمة المرور", "إدخال كلمة المرور", "VbScript")
```

```
If password <> "VbScript" then history.go (0)
```

```
else
```

```
Msgbox password & " كلمة المرور هي "
```

```
Msgbox yourname & " مرحبا بك "
```

```
document.location=" index.htm"
```

```
End if
```

```
</script>
```