أكواد منوعة ومبتكرة للمبرمجين بالفيجوال بيسك

الجزء الأول

تأليف وإعداد / عبدالله خضر عبدالله الهوساوي

E-Mail : prog909@yahoo.com

بسم الله الرحمن الرحيم

هذه مجموعة أكواد مهمة وفنية وتسهل الكثير من الأعمال البرمجية في الفيجوال بيسك ، وهي من تصميمي ، أرجو أن ينتفع بها مبرمجينا الأعزاء ، هي هكذا تباعا :

(دالة وظيفية لتفصيل العدد في مراتبه بإضافة الفواصل)

حيث الإسم الإفتراضي لهذه الدالة يكون : **NDT**
حيث يوضع العدد أو المتغير العددي بين قوسين بعدها أثناء كتابتها في النص البرمجي

شرح مبسط عن مخرجات الدالة :

: مثلا إدخال الآتي :
العدد : 1233948
Print Ndt(1233948)

فتخرج الدالة هذه النتيجة :
1,233,948

ملاحظة مهمة : هذا الكود لهذه الدالة يوضع في الفورم في منطقة الإعلانات العامة للفورم الـ Declarations ، أو في ملف ميديول Moduel تابع لمشروع البرنامج .

نص الكود :

```
Function Ndt(Number As Variant) As String

Z2$ = Format(Number, "Standard")

Ndt = Left$(Z2$, Len(Z2$) - 3)

End Function
```

(دالة وظيفية لإعطاء مقدار النسبة المئوية بين عددين)

حيث الإسم الإفتراضي لهذه الدالة يكون : **Perc**

حيث يوضع العددين أو المتغيرين العدديين بين قوسين بعدها أثناء كتابتها في النص البرمجي

شرح مبسط عن مخرجات الدالة:

مثلا إدخال الآتي:
العدد الكلي أولا : 20
العدد الجزئي ثانيا : 5
Print Perc(20,5)

فتخرج الدالة هذه النتيجة:
25
أي 25%

ملاحظة مهمة : هذا الكود لهذه الدالة يوضع في الفورم في منطقة الإعلانات العامة للفورم الـ Declarations ، أو في ملف ميديول Moduel تابع لمشروع البرنامج .

نص الكود :

```
Function Perc(N1 As Variant, N2 As Variant) As Double

If N2 <= N1 Then

N3 = (N2 / N1) * 100

Perc = Round(N3, 2)

Else

Perc = 0

End If

End Function
```

(دالة وظيفية لتحديد مقدار عدد من نسبة مئوية معطاة)

حيث الإسم الإفتراضي لهذه الدالة يكون **PercN** :

حيث يوضع العددين أو المتغيرين العدديين بين قوسين بعدها أثناء كتابتها في النص البرمجي

شرح مبسط عن مخرجات الدالة:

مثلا إدخال الآتي:
العدد النسبي المئوي المُراد أولا : 50 ، أي 50%
العدد الكلي ثانيا : 30
Print PercN(50,30)

فتخرج الدالة هذه النتيجة:
15

ملاحظة مهمة : هذا الكود لهذه الدالة يوضع في الفورم في منطقة الإعلانات العامة للفورم الـ Declarations ، أو في ملف ميديول Moduel تابع لمشروع البرنامج .

نص الكود :

```
Function Perc(N1 As Variant, N2 As Variant) As Double

If N2 <= N1 Then

N3 = (N2 / N1) * 100

Perc = Round(N3, 2) """ 3

Else

Perc = 0

End If

End Function
```

(دالة وظيفية لإضافة أصفار إلى يسار العدد)

حيث الإسم الإفتراضي لهذه الدالة يكون **AddZero** :

حيث يوضع العدد أو المتغير العددي بين قوسين بعدها أثناء كتابتها في النص البرمجي.

شرح مبسط عن مخرجات الدالة :

مثلا إدخال الآتي:
عدد خانات الأصفار : 4
العدد نفسه : 33
Print AddZero(4,33)

فتخرج الدالة هذه النتيجة:
0033

هذه الدالة مهمة بخصوص أرقام الفواتير أو المستندات في المعاملات التجارية مثلاً .

ملاحظة مهمة : هذا الكود لهذه الدالة يوضع في الفورم في منطقة الإعلانات العامة للفورم الـ Declarations ، أو في ملف ميديول Moduel تابع لمشروع البرنامج .

نص الكود :

```
Function AddZero(ZerosAdded As Integer, Number As Integer) As String

Z = ZerosAdded

M$ = CStr(Int(Number)): L = Len(M$)

C = Z - L

If L > Z Then Q$ = "" Else Q$ = String$(C, "0")

AddZero = Q$ + M$

End Function
```

(دالة وظيفية لمعرفة إسم نوع الملف (إسم الإمتداد))

حيث الإسم الإفتراضي لهذه الدالة يكون **Extn** :

حيث يوضع مسار أو إسم الملف بين قوسين بعدها أثناء كتابتها في النص البرمجي

شرح مبسط عن مخرجات الدالة:

مثلا إدخال الآتي:
مسار أو إسم الملف C:\Doc.txt أو Doc.txt
Print Extn("C:\Doc.txt")

فتخرج الدالة هذه النتيجة:
txt

وفي حال عدم وجود إسم نوع للملف تكون النتيجة هكذا:
Without Extension

ملاحظة مهمة : هذا الكود لهذه الدالة يوضع في الفورم في منطقة الإعلانات العامة للفورم الـ Declarations ، أو في ملف ميديول Moduel تابع لمشروع البرنامج .

نص الكود :

```
Function Extn(FileName As String) As String

LF = Len(FileName): S = InStr(1, FileName, ".")

F$ = UCase$(Mid$(FileName, S + 1, LF - S))

If S = 0 Then F$ = "( Without Extension )"

Extn = F$

End Function
```

(دالة وظيفية لتحميل جميع البيانات دفعة واحدة من ملف تسلسلي)

حيث الإسم الإفتراضي لهذه الدالة يكون **GetAll** :

حيث يوضع مسار أو عنوان الملف بين قوسين بعدها أثناء كتابتها في النص البرمجي

شرح مبسط عن مخرجات الدالة :

مثلا إدخال الآتي: إسم ومسار الملف
X$=GetAll("C:\Document")
Text1=X$
أو
Text1= GetAll("C:\Document")

فتخرج الدالة جميع مايحتويه الملف من بيانات ، ويمكن أن توضع البيانات في تيكست TextBox أو ليبل Label

ملاحظة مهمة : هذا الكود لهذه الدالة يوضع في الفورم في منطقة الإعلانات العامة للفورم الـ Declarations ، أو في ملف ميديول Moduel تابع لمشروع البرنامج .

نص الكود :

```
Function GetAll(FileName As String) As String

Open FileName For Append As 1: Close

If FileLen(FileName) <> 0 Then

Open FileName For Input As 1

P$ = Input$(LOF(1), 1)

Close

Else

P$ = ""

End If

GetAll = P$

End Function
```

(دالة وظيفية لتحويل التاريخ الميلادي إلى هجري)

حيث الإسم الإفتراضي لهذه الدالة يكون **HjDt** :

حيث توضع المتغيرات العددية بهذا الترتيب في الدالة:
رقم السنة الميلادية أولا
رقم الشهر الميلادي ثانيا
رقم اليوم من الشهر الميلادي ثالثا

شرح مبسط عن مخرجات الدالة :

مثلا إدخال الآتي : التاريخ الميلادي 2010 – 8 - 5
2010
8
5

Print HjDt(2010,8,5)
فتخرج الدالة هذه النتيجة :
25-8-1431

ملاحظة مهمة : هذا الكود لهذه الدالة يوضع في الفورم في منطقة الإعلانات العامة للفورم الـ Declarations ، أو في ملف ميديول Moduel تابع لمشروع البرنامج .

نص الكود :

```
Function HjDt(GY, GM, GD As Integer) As Date

D = DateSerial(GY, GM, GD) 'A

VBA.Calendar = vbCalHijri

HjDt = D

End Function
```

(إجراء توليد صوت أو نغمة مع تحديد عدد تكرار الصوت)

حيث الإسم الإفتراضي لهذه الدالة يكون (X1 , X2 , X3) **SOUND** : ولها 3 متغيرات
حيث X1 رقم يحدد مستوى النغمة ، و X2 رقم يحدد قوة صوت النغمة ، و X3 رقم يحدد عدد التكرار للصوت الخارج .

شرح مبسط عن عمل وإستدعاء الإجراء :
مثلا إريد توليد صوت أو نغمة:
Call Sound (250 , 50 , 1)

في البداية يكتب هذا الكود من مكتبة API

ملاحظة مهمة : في البرنامج لابد أن يكتب هذا الكود الأساسي لهذه الدالة في الفورم في أعلى منطقة الإعلانات العامة للفورم الـ Declarations ، أو في أعلى ملف ميديول Moduel تابع لمشروع البرنامج ، وهذا الكود يخص تنشيط أحد إجراءات API

```
Public Declare Function Beep Lib "kernel32"(ByVal dwFreq As Long, ByVal dwDuration As Long) As Long
```

ثم نص الكود الثاني :

إنتبه جيداً : يجب أن يُلحق الإسم الإفتراضي في البرنامج لهذا الإجراء بالكلمة **Call**

هكذا مثلا : Call Sound (250 , 50 , 1)

ملاحظة مهمة : هذا الكود الآتي لهذه الدالة يوضع في الفورم في منطقة الإعلانات العامة للفورم الـ Declarations ، أو في ملف ميديول Moduel تابع لمشروع البرنامج .

نص الكود الثاني :

```
Sub Sound(ToneNum As Variant, LenghNum As Variant, RepeatNum As Variant)

A = Val(ToneNum)
If A < 32 Or A > 160000 Then OK = 1

B = Val(LenghNum): If B = 0 Then B = 1
If B > 999 Then OK = 1

C = Val(RepeatNum): If C = 0 Then C = 1



                    ' تابع '
```

```
                    ' تابع '

If OK <> 1 Then

For I = 1 To C

Call Beep(A, B) ' 3

Next

End If

End Sub
```

(دالة معرفة عدد مرات تشغيل البرنامج)

قد تريد يوماً ما أن تعطي أشخاصاً نسخة تجريبية Trial من برنامجك الذي صنعته بالفيجوال بيسك ، يمكنك ذلك عن طريق تحديد عدد مرات التشغيل للبرنامج ومن ثم يمتنع البرنامج عن التنفيذ مطلقاً ، طبعاً هذه هي الطريقة المثلى للتسويق المحكم لأحد برامجك المهمة والجدية ، إليك الطريقة وهي سهلة:

حيث الإسم الإفتراضي لهذه الدالة يكون GetRunNum :

ملاحظة مهمة : هذا الكود لهذه الدالة يوضع في الفورم في منطقة الإعلانات العامة للفورم الـ Declarations ، أو في ملف ميديول Moduel تابع لمشروع البرنامج .

نص الكود :

```
Function GetRunNum() As String

retvalue = GetSetting("A", "0", "Runcount")
GD$ = Val(retvalue) + 1

SaveSetting "A", "0", "RunCount", GD$

GetRunNum = GD$

End Function
```

كيفية إستخدام هذه الدالة أن تكتبت مثلاً :

A$=GetRunNum()
لابد من القوسين الفارغين الملتصقين ()

ثم:
B=VAL(A$)

تستطيع بكل بساطة كتابة الأمر التالي في الفورم Form في الحدث Load :

نص الكود :

```
Private Sub Form_Load ()

A$=GetRunNum()
B=VAL(A$)
IF B=5 THEN END

End Sub
```

حيث B = عدد المرات المطلوبة لعدد تشغيل البرنامج

والحكمة من كتابة هذا الأمر في الحدث Load هو أن هذا الحدث يكون فاعلاً في أول تشغيل البرنامج أي أنه بوابة الفحص للدخول وبداية التشغيل، وهو ضروري من هذه الناحية.

(كود يسمح بكتابة الأرقام فقط في مربع إدخال النصوص Text)

بعد إضافة الأداة Text إلى الفورم Form ، يكتب الكود الآتي في الحدث KeyPress للكائن Text كالآتي :

نص الكود :

```
Private Sub Text1_KeyPress(KeyAscii As Integer)

P$ = "1234567890-." + Chr$(8) + Chr$(46) + Chr$(13)

M$ = Chr$(KeyAscii)

If InStr(P$, M$) = 0 Then KeyAscii = 0

End Sub
```

ملاحظة مهمة : ارقام الكود آسكي : 8=زر إلغاء الخلف ، 48=زر إلغاء الأمام ، 13=زر الإدخال
وأضفت هنا في الحزمة لإحتمال إستخدامها في مربع النصوص لحذف الأخطاء مثلاً أو غيره..

ويمكن إضافة مايراد السماح بطباعته غير الأرقام في المتغير P$

وفي الختام أتمنى أن يفيد هذا الموضوع من يهمه الأمر من المبرمجين بالفيجوال بيسك ،
وإلى مجموعة أكواد مبتكرة أخرى إن شاء الله ،،

تأليف وإعداد / عبدالله خضر عبدالله الهوساوي

E-Mail : prog909@yahoo.com