

بِسْمِ اللَّهِ الرَّحْمَنِ الرَّحِيمِ

التعامل مع الحروف في Visual Basic 6.0



الكاتب: غسان الجزار

البريد الإلكتروني : virusnake@mail2world.com

صفحة الويب : www.geocities.com/virusnake_code

{ يمكنك مراسلتي لأي توضيح أو استفسار على أي فقرة من فقرات الكتاب }

ملاحظة: هذا الكتاب مجاني و يمنع بيعه أو استغلاله تحت الملاحقة القانونية.

حقوق النشر محفوظة ©

* البرامج المذكورة في هذا الكتاب من "Microsoft Visual Basic 6.0"، "Windows" و "Microsoft Visual C++ 6.0" جميعها علامات تجارية لها ملكيتها و الكاتب يحترمها .
* هذا الكتاب مجاني للجميع و يمنع منعاً باتاً بيع أو اقتباس أي جزء من هذا الكتاب إلا بأذن خطي من الكاتب .

* كل الفكر الموجودة داخل الكتاب فكر خاصة بالكاتب أرجو عدم تطبيق الفكر حرفياً لأنها كتبت للتوضيح ، أرجو أن { تحرر عقلك و تبتكر فكر جديدة } .

إهداء

أهدي هذا الكتاب إلى كل شخص
أحب هذه اللغة و تعلق بها و كأنها
خلية من خلايا جسمه إلى كل
شخص شعر بلذة ما يتعلمه سواء
كان عربي أم غربي و في شتى
أنحاء العالم.

مقدمة

* السلام عليكم ورحمة الله وبركاته ، عزيزي القارئ في الحياة عوائق كثيرة يجب اجتنابها ، أرجو أن تكون العوائق التي اعترضت طريقك في تصميم البرامج و المشاريع أن تحل عند قراءتك لهذا الكتاب .

* لا أريد أن تكون قراءتك لهذا الكتاب (فرض) عليك من قبل أحد بل يجب أن تشعر في لذة ما تقرأه في هذا الكتاب أو غيره من الكتب في شتى أنحاء العالم مثل ما شعرت أنا عندما قرأت أول كتاب يتحدث عن البرمجة الكائنية .

* سأقدم في هذا الكتاب كل ما يتعلق عن الحروف و الرموز و ألحقت مع كل مثال مجلد يحوي تطبيق عملي عنه .

* تقوم فقرات الكتاب عن شرح بسيط و سهل للاستيعاب السريع .

* أرجو مراعاتي في الأخطاء النحوية و الإملائية لسرعتي في الكتابة .

* هذا الكتاب كتب للمتوسطين و المحترفين أي لمن أخذ فكرة عن Visual Basic .

المتغير الحرفي :

المتغير الحرفي إنه متغير يحجب في الذاكرة مساحة بايت لكل رمز أو حرف قد تعادل ٢ جيجا بايت ويتم الإعلان عنه باستخدام Dim و بعدها المتغير الذي ستسند القيمة له و ليكن X و نعرف جميعنا أننا نستطيع أن نضع As Integer أو As Long أو As Single للأرقام بشتى أنواعها لكن للحروف نستخدم As String ، ثم نسند القيمة لمتغير ولكن عند إسناد قيمة حرفية للمتغير لا تنسى أن تضعها في إشارة التنقيص (" ") .
مثال:

```
Private Sub Command1_Click () 'Code 01.zip
Dim X As String
X = "I Like Visual Basic"
Print X
End Sub
```

و يمكننا مباشرة أن نتحكم بعدد الحروف التي يحملها المتغير:
مثال:

```
Private Sub Command1_Click () 'Code 02.zip
Dim X As String * 13
X = "I Like Visual Basic"
Print X
End Sub
```

نلاحظ أنه لم يطبع سوى "I Like Visual" .

وفي لغة C++ نكتب :

```
#include "stdafx.h"
#include "iostream.h"

int main()
{
    char x[18]="I Like Visual C++";
    cout<<x<<endl;
}
```

مباشرةً نحدد عدد
الحروف

ملاحظة: عند تحديد عدد الحروف التي سيحملها المتغير يبدأ العد من الصفر و ليس من الواحد و أيضاً المسافة تحسب حرف و لها قيمة مثل بقية الحروف .

الدالة Len لتحديد الطول :

صيغتها العامة :

Len(Expression)

نستخدم عادةً لحساب طول متغير حرفي الأداة Len مرفقة مع أسم المتغير .

مثال:

```
Private Sub Command1_Click () 'Code 03.zip
Dim X As String
X = "I Like Visual Basic"
Print Len(X)
End Sub
```

نلاحظ طباعة "19".

و يمكننا استخدامها مثلاً بأن نجعل المستخدم بعدم إدخال أكثر من عدد معين من الأعداد أو الحروف .
مثال:

```
Private Sub Command1_Click () 'Code 04.zip
Dim X As String
X = Text1.Text
If Len(X) >= 8 Then
MsgBox "Error!"
Text1.Text = ""
End If
End Sub
```

و يمكنك مباشرة تغيير خاصية MaxLength في Textbox وتدخل عدد الحروف التي تقبلها .

الدالة InStr :

صيغتها العامة:

InStr([Start], [String1], [String2], [Compare As VbCompareMethod])

يمكننا بواسطة هذه الدالة معرفة موقع حرف في النص .
مثال:

```
Private Sub Command1_Click () 'Code 05.zip
Dim X As String
X = "I Like Visual Basic"
Print InStr(1, X, "V")
End Sub
```

سيطبع العدد ٨ و هو موضع الحرف V في النص .
أما الدالة InStrRev فهي مثل الدالة InStr لكنها تبدأ من نهاية القيمة المرسلة لها.
صيغتها العامة:

*InStrRev(StringCheck As String, StringMatch As String, [Start As Long = -1],
[Compare As VbCompareMethod]) As Long*

معامل الربط & :

صيغتها العامة:

[String1] & [String2]

تعد أداة الربط من أهم الأدوات لأنها تساعد على الربط بين نصين .

يمكن استخدامها للترحيب بأحد الزوار أو المستخدمين.
مثال:

```
Private Sub Command1_Click () 'Code 06.zip
Dim X As String
X = Text1.Text
Print "المشترك الأول" & X
End Sub
```

سأوضح الفرق بين معامل الربط & و معامل الجمع + عند الربط بين جملتين نصيتين فإن الأدوات تقومان بنفس العمل إلا إذا أردنا أن نربط بين جملة نصية و رقم بأداة الجمع + فلا بد بظهور رسالة الخطأ الغير مرغوب فيها.
مثال:

```
Private Sub Command1_Click()
Dim X As String
X = "Visaul Basic "
Print X + 6 'Error!
End Sub
```

لكن يمكننا إجراء التعديل اللازم أي تبديل معامل الجمع + بمعامل الربط & .
مثال:

```
Private Sub Command1_Click()
Dim X As String
X = "Visaul Basic "
Print X & 6
End Sub
```

نلاحظ أنه سيطلع عبارة Visual Basic 6 دون أي خطأ ، لكن يمكننا أن نضع معامل الجمع + بشرط وضع الرقم بين قوسي التنصيص أي "6" .

الدالة Str\$:

صيغتها العامة:

Str\$(Number)As String

بتحويل الأرقام إلى سلاسل حرفية أي بربط بين رقمين دون جمعهما.
ملاحظة: إذا كنت تريد أن تعرف الفرق بين الدالة Str والدالة Str\$ الدالتان متشابهتان في المضمون لكن الدالة الأولى تعود بالقيمة من نوع Varient أما الثانية فتعود بالقيمة من نوع String مما يؤدي زيادة سرعة الأداء ، إذا وجد من الدالة نسختان أستخدم الدالة ذات الرمز \$.
مثال:

```
Private Sub Command1_Click() 'Code 07.zip
Dim X1, X2 As Long
X1 = 200
X2 = 5
Print Str$(X1) + Str$(X2)
End Sub
```

نلاحظ أنه سيطلع 5 200 بدل من 205 لكن لاحظوا أنه سيضيف مسافة في نهاية كل سلسلة نصية يمكنك استخدام الدالة Cstr بدلاً من الدالة Str\$.

مثال:

```
Private Sub Command1_Click () 'Code 08.zip
Dim X1, X2 As Long
X1 = 200
X2 = 5
Print CStr(X1) + CStr(X2)
End Sub
```

نلاحظ طباعة 2005 دون أي مسافات .

نلاحظ أن المعاملات المنطقية < > = < > ليست حكرًا على الأرقام بل تصلح للحروف أيضاً حيث تكون قيمة الحروف هي الأرقام المقابلة لها في جدول Ascii .
مثال:

```
Private Sub Command1_Click () 'Code 09.zip
Dim X1, X2 As String
X1 = "Basic"
X2 = "BASIC"
Print X1 > X2 'True
Print X1 < X2 'False
Print X1 <> X2 'True
Print X1 = X2 'False
End Sub
```

معامل التشابه Like :

يستخدم معامل التشابه Like لإجراء مقارنة غير دقيقة أو سطحية فمثلاً أنت تريد أن تتدخل قيمة معينة ولا تريد أن تكون سوى أرقام أو حروف فتستطيع تحديد نوع المدخلات و عددها و نطاقها نستخدم الرمز # للأرقام و الرمز ? للحروف و الأرقام و الرمز * لعدد من الحروف و الأرقام .
مثال ١:

```
Private Sub Command1_Click () 'Code 10.zip
Dim X As String
X = "1234567"
If X Like "1#####" Then Print X
End Sub
```

يقوم في التحقق من قيمة المتغير إذا كانت أولها العدد واحد ثم ستة أرقام أيًا كانت سيقوم بطباعته .
مثال ٢:

```
Private Sub Command2_Click () 'Code 10.zip
Dim X As String
X = "ABCD123"
If X Like "A?????" Then Print X
End Sub
```

يقوم في التحقق من قيمة المتغير إذا كانت أولها الحرف A ثم ستة حروف أو أرقام أيًا كانت سيقوم بطباعته .

مثال ٣:

```
Private Sub Command3_Click () 'Code 10.zip
Dim X As String
X = "ABCD123"
If X Like "A*" Then Print X
End Sub
```

يقوم في التحقق من قيمة المتغير إذا كانت أولها الحرف A ثم أي عدد من الحروف و الأرقام سيقوم طباعته .

ويمكنك تحديد مدى الحرف باستخدام القوسين المربعين [] .
مثال ١:

```
Private Sub Command1_Click () 'Code 11.zip
Dim X As String
X = "Visual"
If X Like "[A-Z]*****" Then Print X
End Sub
```

يقوم في التحقق من قيمة المتغير إذا كانت أولها الحرف أي حرف في نطاق A إلى Z ثم خمسة حروف أو أرقام أيأ كانت سيقوم طباعته .

ويمكنك إسناد حرف أو مجموعة حروف أو أرقام باستخدام الرمز ! .
مثال ٢:

```
Private Sub Command2_Click () 'Code 11.zip
Dim X As String
X = "Visual 6"
If X Like "*****[!A-Z]" Then Print X
End Sub
```

يقوم في التحقق من قيمة المتغير إذا كانت آخرها أي حرف أو رمز أو رقم ما عدا الأحرف في النطاق A إلى Z الكبيرة و أولها ستة حروف أو أرقام مهما كانت ثم يقوم في طباعتها .

الدالة Replace للاستبدال :

صيغتها العامة:

Replace(Expression As String, Find As String, Replace As String, [Start As Long = 1], [Count As Long = -1], [Compare As VbCompareMethod]) As String

لن تجد أسرع من هذه الدالة لاستبدال النصوص و أنا شخصياً أحبها أكثر من غيرها من الأدوات.
مثال:

```
Private Sub Command1_Click () 'Code 12.zip
Dim X As String
X = Text1.Text
Text1.Text = Replace(X, "Visual", "Microsoft Visual")
End Sub
```

نلاحظ باستبدال كلمة Visual بكلمة Microsoft Visual بسرعة .

أدوات تنسيق الشكل :

نذكر منها الدالة LCase التي تجعل النص الإنكليزي يظهر بالشكل الصغير .
صيغتها العامة:

LCase\$(String)As String

مثال ١ :

```
Private Sub Command1_Click () 'Code 13.zip
Dim X As String
X = "I Like Visual Basic"
Print LCase$(X)
End Sub
```

نلاحظ طباعة الكلمة "i like visual basic"

أما الدالة UCase فهي عكس السابقة نسبياً فهي تقوم بتكبير الحروف بدلاً من تصغيرها.
صيغتها العامة:

UCase\$(String)As String

مثال ٢ :

```
Private Sub Command1_Click () 'Code 13.zip
Dim X As String
X = "I Like Visual Basic"
Print UCase$(X)
End Sub
```

نلاحظ طباعة الكلمة "I LIKE VISUAL BASIC"

من الدوال الغريبة دالة القلب StrReverse و تقوم على قلب النص رأساً على عقب.
صيغتها العامة:

StrReverse (Expression As String) As String

مثال ٣ :

```
Private Sub Command1_Click () 'Code 13.zip
Dim X As String
X = "I Like Visual Basic"
Print StrReverse (X)
End Sub
```

نلاحظ طباعة الكلمة "I ekiL I VsiL aB cisi"

الدالة StrConv أداة تنسيق متكاملة :

صيغتها العامة:

StrConv(String, Conversion As VbStrConv, [LocaleID As Long])

يمكننا من خلال هذه الأداة تكبير و تصغير الحروف و أيضاً تكبير أول حرف.

بإرسال الثابت vbLowerCase للتصغير ، الثابت vbUpperCase للتكبير و الثابت vbProperCase لتكبير أول حرف .
مثال ١:

```
Private Sub Command1_Click () 'Code 14.zip
Dim X As String
X = "I lIkE vIsUaL bAsIc"
Print StrConv(X, vbLowerCase)
End Sub
```

نلاحظ طباعة "i like visual basic"

مثال ٢:

```
Private Sub Command2_Click () 'Code 14.zip
Dim X As String
X = "I lIkE vIsUaL bAsIc"
Print StrConv(X, vbUpperCase)
End Sub
```

نلاحظ طباعة "I LIKE VISUAL BASIC"

مثال ٣:

```
Private Sub Command3_Click () 'Code 14.zip
Dim X As String
X = "I lIkE vIsUaL bAsIc"
Print StrConv(X, vbProperCase)
End Sub
```

نلاحظ طباعة "I Like Visual Basic"

إذا كنت تريد أن تجعل المستخدم أن يدخل كلمة مرور مع عدم مراعاة الحرف الكبيرة من لصغيرة .
هذا المثال سيوضح :

```
Private Sub Command1_Click () 'Code 15.zip
Dim X As String
X = LCase$(Text1.Text)
If X = "ok" Then MsgBox "كلمة المرور صحيحة"
End Sub
```

في هذا المثال قمنا بإعطاء قيمة الـ TextBox لكن بتصغير الحروف و من ثم التحقق من الكلمة المدخلة و هي صغيرة في المتغير مع الكلمة التي نحن بالأصل وضعناها صغيرة .

ملاحظة: إذا كنت جاد في برنامجك فيمكنك إضافة الكلمة المحبوبة Option Compare Text ووضعها في ساحة الإعلانات العامة .
مثال:

```
Option Compare Text

Private Sub Command1_Click () 'Code 16.zip
Dim X As String
X = Text1.Text
If X = "ok" Then MsgBox "كلمة المرور صحيحة"
End Sub
```

نلاحظ أن المستخدم إذا أدخل OK أو ok تكون الكلمة صحيحة .

ملاحظة: إذا كنت تريد أن تكتب في Label عبارتان فوق بعضهما أي كأنك ستنزل سطر في محرر النصوص أستخدم الكلمة المحجوبة VbCrLf ستفي بالغرض.
مثال:

```
Private Sub Command1_Click () 'Code 17.zip
Dim X1, X2 As String
X1 = "Microsoft"
X2 = "Visual Basic 0.6"
Label1.Caption = X1 & vbCrLf & X2
End Sub
```

الدوال (Asc - AscW) و (Chr\$ - ChrW\$):

صيغهم العامة:

Asc(String As String) As Integer
AscW(String As String) As Integer
Chr\$(CharCode As Long) As String
ChrW\$(CharCode As Long) As String

الدالتان Asc و AscW تعودان بالرقم المقابل للحرف في جدول Ascii مع اختلاف في السرعة بينهما .
مثال:

```
Private Sub Command1_Click () 'Code 18.zip
Print Asc("ع")
Print AscW("ع")
End Sub
```

الأولى ستطبع ٢١٩ و الثانية ستطبع ١٥٩٤ .
إذا كنت تريد أن تتراجع عن ما فعلته في المثال السابق فالدالتان Chr\$ و ChrW\$ تفيان في الغرض .
مثال:

```
Private Sub Command2_Click () 'Code 18.zip
Print Chr$(219)
Print ChrW$(1594)
End Sub
```

لكن أعيد و أكرر مع اختلاف في السرعة .
ملاحظة: الدالة Chr\$ أم ChrW\$ يمكنك من خلالها خداع المترجم أي عندما تطبع كلمة كنت قد وضعتها في قوسي التنصيص لا يظهر قوسا التنصيص أما إذا كنت تريد أن تظهرهما لا تستطيع إلا من خلال هاتان الدالتان .
مثال:

```
Private Sub Command1_Click ()
Dim X As String
X= (" "Visual Basic" ") 'Error!
Print X
End Sub
```

أما إذا كتبنا الرقم الذي يدل على قوسا التنصيص في جدول الـ Ascii .

مثال:

```
Private Sub Command1_Click () 'Code 19.zip
Dim X As String
X = Chr$(34) & "Visual Basic" & Chr$(34)
Print X
End Sub
```

إذا كنت تريد أن تعرف رموز جميع الأرقام و الحروف هذا الكود سيفي في الغرض .

```
Private Sub Form_KeyPress(KeyAscii As Integer) 'Code 20.zip
Form1.Cls
Label1.Caption = "Asc [ " & Chr$(KeyAscii) & " ] <=> " & " ( " & (KeyAscii) & " ) "
End Sub
```

دالة التنسيق الغير قياسية : Format\$

صيغتها العامة:

*Function Format\$(Expression, [Format], [FirstDayOfWeek As VbDayOfWeek],
[FirstWeekOfYear As VbFirstWeekOfYear]) As String*

لا أقصد بكلمة (غير قياسية) أنها ذم لا سمح الله إنما هي دالة تستخدم تنسيقات خاصة مثل الرمز @ لحرف أو رقم أو مسافة و الرمز & الذي يمثل حرف أو رقم أو لاشيء .
مثال:

```
Private Sub Command1_Click () 'Code 21.zip
Dim X1, X2 As String
X1 = "CreatedByMicrosoft"
X2 = "Version104236"
Print Format(X1, "@@@@@@-@@-@@@@@@@@.")
Print Format(X2, "@@@@@@@@:&.&.&.&")
End Sub
```

سيطبع التالي :

Created-By-Microsoft
Version:1.04.236

عفواً نسيت شيء مهم سنعود إلى الدالة Len ثانية فبدلاً من استخدام قوسي التنصيص الفارغين للتحقق من خلو النص أستخدم الدالة Len فهي أسرع منهما .
مثال:

```
Private Sub Command1_Click () 'Code 22.zip
If Text1.Text = "" Then Print "Type Something"
End Sub

Private Sub Command2_Click () 'Code 22.zip
If Len(Text1.Text) = 0 Then Print "Type Something"
End Sub
```

الدالة Len أسرع بمرتين تقريباً من قوسي التنصيص .

دوال الحجب :

نذكر منها الدوال Left\$ و Right\$ و Mid\$ و LTrim\$ و RTrim\$ و Trim\$ سنشرح كل واحدة على حدة .

١ - دالة الحجب إلى اليسار Left\$:

صيغتها العامة:

Left\$(String As String, Length As Long) As String

تقوم هذه الدالة بقص منطقة من يسار النص بمقدار معين .
مثال ١ :

```
Private Sub Command1_Click () 'Code 23.zip
Dim X As String
X = "MicrosoftVisualBasic"
Print Left$(X, 9)
End Sub
```

نلاحظ طباعة "Microsoft" .

٢ - دالة الحجب إلى اليمين Right\$:

صيغتها العامة:

Right\$(String As String, Length As Long) As String

تقوم هذه الدالة بعكس ما تقومه أختها Left\$ أي قص النص من اليمين بمقدار معين .
مثال ٢ :

```
Private Sub Command2_Click () 'Code 23.zip
Dim X As String
X = "MicrosoftVisualBasic"
Print Right$(X, 5)
End Sub
```

نلاحظ طباعة "Basic" .

٣ - دالة الحجب من الوسط Mid\$:

صيغتها العامة:

Mid\$(String As String, Start As Long, [Length]) As String

تقوم هذه الدالة بقص النص من أوله بمقدار معين مع تعيين بداية و نهاية القص .
مثال ٣ :

```
Private Sub Command3_Click () 'Code 23.zip
Dim X As String
X = "MicrosoftVisualBasic"
Print Mid$(X, 10, 6)
End Sub
```

نلاحظ طباعة "Visual" .

لم تنتهي من هذه الدالة فنستطيع أن نتعامل معها على أنها تصريح أو بيان أي (Statement) فيمكنها أن تقص و تستبدل أيضاً تصور يا رعاك الله .
مثال:

```
Private Sub Command1_Click () 'Code 24.zip
Dim X As String
X = "MicrosoftlausiVBasic"
Mid$(X, 10, 6) = "Visual"
Print Mid$(X, 10, 6)
End Sub
```

نلاحظ أنها قصت كلمة lausiV وبدلتها بـ Visual ومن ثم قصت كلمة Visual وطبعتها.

٤- دالة حذف المسافة من اليسار LTrim\$:

صيغتها العامة:

LTrim\$(String As String) As String

تقوم هذه الدالة بحذف جميع المسافات على يسار النص .
مثال ١ :

```
Private Sub Command1_Click () 'Code 25.zip
Dim X As String
X = " Visual Basic "
Label1.Caption = LTrim$(X)
End Sub
```

٥- دالة حذف المسافة من اليمين RTrim\$:

صيغتها العامة:

RTrim\$(String As String) As String

تقوم هذه الدالة بحذف جميع المسافات على يمين النص .
مثال ٢ :

```
Private Sub Command1_Click () 'Code 25.zip
Dim X As String
X = " Visual Basic "
Label1.Caption = RTrim$(X)
End Sub
```

٦- دالة حذف المسافة Trim\$:

صيغتها العامة:

Trim\$(String As String) As String

إذا كان أدخل المستخدم مسافة من دون أن يدري فهذا لن يتطابق مع القيمة المخزنة في البرنامج
أستخدم هذه الدالة لحذف المسافات على طرفي النص .
مثال ٣ :

```
Private Sub Command1_Click () 'Code 25.zip
Dim X As String
X = " Visual Basic "
Label1.Caption = Trim$(X)
End Sub
```

الدالتان Split و Join :

الدالة Split تقوم بفصل الكلمات عن بعضها البعض إذا كان بينها حرف أو رمز.
صيغتها العامة:

Split(Expression As String, [Delimiter], [Limit As Long = -1], [Compare As VbCompareMethod])

مثال ١:

```
Private Sub Command1_Click () 'Code 26.zip
Dim X As String
Dim XSplit() As String
Dim I As Integer

X = "Micrisoft-Visual-Basic"

XSplit = Split(X)
XSplit = Split(X, "-")

For I = 0 To UBound(XSplit)
Print XSplit(I)
Next I
End Sub
```

قمنا الآن بفص كل كلمة واقعة إمام أشار "-" عن الكلمة الأخرى.

الدالة Join ترجع الكلمة كما كانت عليه أي بعد فصلها عن بعضها.
صيغتها العامة:

Join(SourceArray, [Delimiter]) As String

مثال ٢:

```
Private Sub Command2_Click () 'Code 26.zip
Dim X As String
Dim XSplit() As String
Dim XJoin As String

X = "Micrisoft-Visual-Basic"

XSplit = Split(X)
XSplit = Split(X, "-")

XJoin = Join(XSplit)
XJoin = Join(XSplit, "-")

Print XJoin
End Sub
```

لقد عادت الكلمة كما كانت .

في النهاية يمكننا تلخيص ما كتبناه في هذا الجدول :

عملها	الدالة
لتحديد الطول النص	Len
لمعرفة موقع حرف في النص أو البحث عنه	InStr
لربط بين نصين	&
لتحويل النصوص إلى سلاسل رقمية	Str\$
لتحويل النصوص إلى سلاسل رقمية بدون مسافة	CStr
إجراء مقارنة غير دقيقة	Like
لاستبدال النص بنص آخر	Replace
لتصغير الحروف	LCase\$
لتكبير الحروف	UCase\$
لقب النص	StrReverse
لتكبير و تصغير و تكبير أول حرف في النص	StrConv
للنزول سطر	VbCrLf
لإرجاع الحرف إلى الرقم المقابل له في جدول Ascii	Asc , AscW
لإعادة الرقم إلى حرفه	Chr\$, ChrW\$
لتنسيقات خاصة كثيرة	Format\$
لقص حروف من اليسار	Left\$
لقص حروف من اليمين	Right\$
لقص حروف من الوسط	Mid\$
لحذف المسافة من اليسار	LTrim\$
لحذف المسافة من اليمين	RTrim\$
لحذف المسافة من الطرفين	Trim\$
لقطع الكلمات عن بعضها	Split
لتركيب الكلمات بعد القطع	Join

النهائية

طبعة أولى في : ٢٧/٤/٢٠٠٥