المحاضرة السابعة

استعمال الحلقات والمؤقتات

١. حلقة For....Next

تتيح هذه الحلقة تنفيذ مجموعة معينة من الجمل البرمجية عددا محددا من المرات

الصيغة العامة لها

```
for  variable = start  to    end

  statements to be repeated

next  variable
```

مثال :-

```
dim  i as integer

for i =1    to      4

beep( )

next  i
```

هذه الحلقة تقوم بإصدار أربعة أصوات تنبيه متتالية حيث أن الحلقة أعلاه هي المرادف لكتابة الجملة beep أربع مرات

```
beep()

beep()

beep()

beep()
```

المتغير المستخدم في الحلقة هو i وهو يعرف بالعداد canter ويجب ان يكون من النوع العددي وفي كل مرة يتم فيها تنفيذ الحلقة تزداد قيمته بمقدار واحد حتى الوصول إلى نهاية الحلقة ،حيث أيضاً (1)تمثل بداية الحلقة start و(4) تمثل نهاية الحلقة end .

مثال: باستخدام لغة VB.Net اكتب برنامج لطباعة النص التالي:

Line1

Line2

Line10

في أداة مربع نص TextBox1

مرحلة التصميم :-

a. إضافة أداة مربع نص TextBox1 مع جعل خاصية Multi Line لمربع النص True

b. إضافة زر أمر Button1 مع جعل خاصية text لزر الأمر "طباعة"

مرحلة التصميم :-

كتابة الكود التالي في زر الأمر Button1

```
Private Sub Button1_Click(ByVal sender As System.Object, ByVal e As System.EventArgs)
Handles Button1.Click
        Dim i As Integer
```

```
    Dim x As String
    x = Chr(13) + Chr(10)
    For i = 1 To 10
        TextBox1.Text = TextBox1.Text & "Line" & i & x
    Next
End Sub
```

ملاحظة:-

الأداة & تستخدم للدمج والجملة Chr(13) + Chr(10) وظيفتها نقل مؤشر الفأرة لسطر جديد

المتغير العداد في الحلقة for---next يمكن استخدامه لإنشاء حلقة بنمط مختلف عن 1,2,3,4

حيث يمكنك تحديد قيمة مختلفة لـــــــ start في الحلقة ثم استعمال الكلمة الأساسية step لزيادة العداد بفواصل وقيمة مختلفة.

مثلاً الشفرة

```
dim  i  as integer
dim wrap as string
wrap = chr(13)&chr(10)
for i = 5 to 25 step 5
textbox1.text = textbox1.text & "Line" & i  &wrap
next  i
```

وناتج هذه الشفرة هـــــو

Line5

Line10

Line15

Line20

Line25

كما يمكنك أيضاً تحديد قيم عشرية في الحلقة

مثلا

```
dim  i  as single
dim wrap as string
wrap = chr(13)&chr(10)
for i = 1 to 2.5 step 0.5
textbox1.text = textbox1.text & "Line" & i  &wrap
next  i
```

الجملة exit for

تستخدم مع الحلقة for --- next لإنهاء تنفيذ الحلقة التكرارية

مثلاً:

```
dim i as integer
dim inp_name as string
for  i = 1  to  10
inp_name = inputbox("enter your name or type done to quit.")
if inp_name = "done"    then  exit for
textbox1.text = inp_name
next  i
```

الحلقة أعلاه تطلب من المستخدم كتابة 10 أسماء وتعرضها الواحد تلو الآخر في مربع النص إلا إذا كتب المستخدم الكلمة done

٢. حلقات do

كبديل عن الحلقة for----next يمكنك كتابة حلقة do لتنفيذ مجموعة من الجمل إلى أن يصبح شرط معين صحيحاً .

الصيغة العامة الأكثر شيوعا

```
do while condition
    statements
loop
```

مثال :

الحلقة التالية تطلب إدخالاً من المستخدم وتعرض ذلك الإدخال في مربع نص إلى أن تكتب الكلمة done في مربع الإدخال

```
dim i  as  string
do while i <> "done"
i = inputbox("enter your name or type done to quit ")
if  i  <> "done"  then textbox1.text = i
loop
```

كما يمكن ان يكون اختبار الشرط في حلقة do--- while عند نهاية الحلقة

وذلك كما يلي :-

```
dim i as string
i  = inputbox("enter your name or type done to quit")
if  i  <> "done"  then textbox1.text = i
loop while  i  <> "done"
```

استعمال الكلمة until في الحلقات do

يمكن استعمال الكلمة until في الحلقات do للتكرار إلى أن يصبح الشرط صحيحاً

حيث يمكنك استخدام الكلمة until في أعلى أو أسفل الحلقة do لاختبار الشرط تماماً مثل الكلمة while

مثال :

```
dim i as string
do
i = inputbox("enter your name or type done to quite ")
if i <> "done" then textbox1.text = i
loop until i = "done"
```

مـــــــلاحـــــــظـــــة : -

الحلقة التي تستخدم الكلمة until مشابهة للحلقة التي تستعمل الكلمة while **ماعدا ان شرط الاختبار** يحتوي عادة على المعامل المعاكس (=) يساوي مقابل المعامل (< >) لا يساوي

أدارة المؤقت Timer

تستعمل هذه الأداة لتنفيذ مجموعة من الجمل لفترة زمنية محددة أو عند فواصل زمنية محددة وهي في الأساس ساعة توقيت غير مرئية.

رغم إن المؤقتات لا تكون مرئية إثناء التشغيل إلا إن كل مؤقت يكون مقترنًا بإجراء حدثي يشتغل كلما انقضى الفاصل الزمني المحدد مسبقاً.

ومن خصائص أداة التوقيت

١. الخاصية Interval تستخدم لضبط الفاصل الزمني المؤقت.

٢. الخاصية Enabled تستخدم لتنشيط المؤقت عندما تأخذ القيمة True

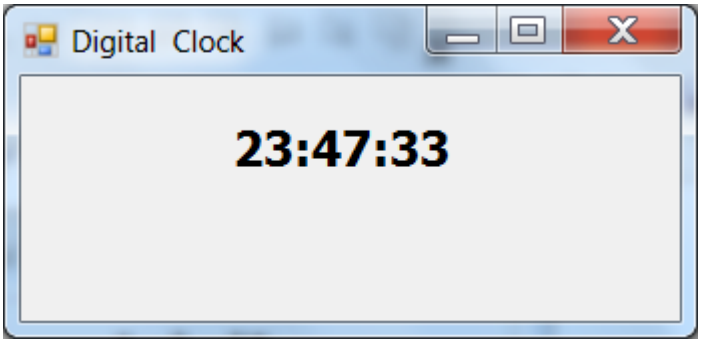
مثال :

اكتب برنامج لإنشاء ساعة رقمية باستخدام أداة المؤقت Timer

مرحلة التصميم

أضف العناصر التالية إلى النموذج مع ضبط خصائصها حسب الجدول التالي

الكائن	الخاصية	القيمة
Form1	Text	Digital Clock
Timer1	Enabled	True
	Interval	١٠٠٠
Label1	Text	فارغة



مرحلة البرمجة :-

اكتب الكود التالي في الإجراء الحدثي للمؤقت

```
Private Sub Timer1_Tick(ByVal sender As System.Object, ByVal e As System.EventArgs)
Handles Timer1.Tick
        Label1.Text = TimeString
    End Sub
```

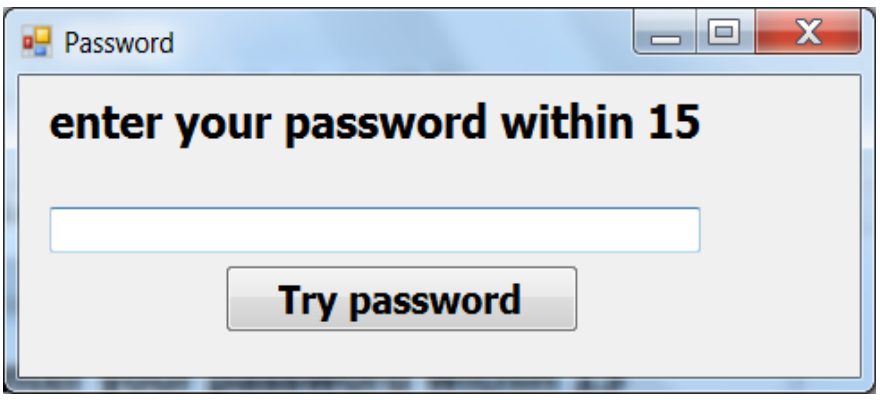
الجملة `Label1.Text = TimeString` تأخذ الوقت الحالي من الساعة الحالية وتعينه للأداة label 1 ولإظهار التاريخ بجانب الساعة استعمل الخاصية `System.DateTime.Now` بدلاً من الخاصية `TimeString`

كما يمكن استعمال مؤقت أيضاً لفرض حد أقصى على الوقت وهو ضبط المؤقت لانتظار فترة من الوقت ثم إما تمكين المستخدم أو منعه من تنفيذ عمل ما

مثال :

اكتب برنامج لضبط حد أقصى لكتابة كلمة المرور

أنشى النموذج التالي:



واضبط خصائص الكائنات كما يلي :-

الكائن	الخاصية	القيمة
Form1	Text	Password
Timer1	Enabled	True
	Interval	١٥٠٠٠
Label1	Text	Enter your password with in 15 seconds
TextBox1	password	*
Button1	Text	Try Password

انقر نقراً مزدوجاً على أداة المؤقت ثم اكتب الجملة التالية:

msgbox("sorry, your time is up.")

ثم أكتب الجمل التالية في الأمر Try Password

```
Private Sub Button1_Click_1(ByVal sender As System.Object, ByVal e As System.EventArgs)
Handles Button1.Click
        If TextBox1.Text = "secret" Then
            Timer1.Enabled = False
            MsgBox("welome to the system")
        Else
            MsgBox("Sorry ,friend idont know you")

        End If
    End Sub
```