الدوارات

التكرار Looping

- تنفيذ مجموعة أوامر عدة مرات متتالية .

أنواع التكرار

- تكرار لعدد محدود من المرات .
- تكرار طالما تحقق شرط ما .

تكرار الأوامر لعدد محدود من المرات For . . . Next

```
For counter variable = start To end [step]

        [ أوامر لتنفيذها داخل التكرار ]

Next [counter variable]
```

١- يبدأ التكرار بتعيين قيمة متغير (عداد) إلى قيمة البداية .
٢- ينفذ الفيجوال الأوامر مرة واحدة .
٣- يزيد العداد بالمقدار الذي يحدده العداد إلى قيمة النهاية المحددة .
٤- إذا وصل العداد للنهاية ينتهي التكرار و إذا لم يصل ينتقل للخطوة (٢)

❖للخروج من التكرار نستخدم العبارة

Exit For

أغلاق التكرار بعبارة Next

- لكل عبارة for توجد عبارة Next لتعيد البرنامج إلى بداية جملة for
- عبارة Next قد تحتوي على إسم العداد أو لا .

مثال

```
Dim intc As integer
For  intc = 1 To 100
        debug.writeline (intc)
Next  intc
```

النتيجة

```
1
2
3
.
.
.
```

تحديد مقدار الزيادة بإستخدام Step

- Step تحدد مقدار زيادة العداد بعد إتمام كل دورة من التكرار .
- عند حذف كلمة Step فإن الزيادة تكون بمقدار واحد .

مثال

```
Dim intc As integer
For  intc = 1 To 100  Step 4
          debug.writeline (intc)
Next  intc
```

النتيجة

```
1
5
9
.
.
.
```

تحديد مقدار الزيادة بإستخدام Step

مثال

```
Dim intc As integer
For  intc = 100 To 1  Step -1
        debug.writeline (intc)
Next  intc
```

النتيجة

```
100
99
98
.
.
.
```

الخروج من التكرار قبل إكتماله

مثال

```
Dim intc As integer
For  intc = 1 To 100
        if intc=50 then
            Exit for
debug.writeline (intc)
Next  intc
```

- في حال تحقق الشرط <condition> فإن التحكم سوف يخرج من For وينتقل إلى الجملة التي تلي الجملة Next

العودة إلى بداية التكرار قبل الوصول إلى العبارة Next

```
For  counter= 1 To 100
        if <expression> then
            continue for
        End if
        أوامر أخرى
Next  counter
```

```
Dim intc As integer
For  intc = 1 To 100
        if intc=50 then
            continue  for
debug.writeline (intc)
Next  intc
```

النتيجة:

١
٢
...
٤٩
٥١
٥٢
...
١٠٠

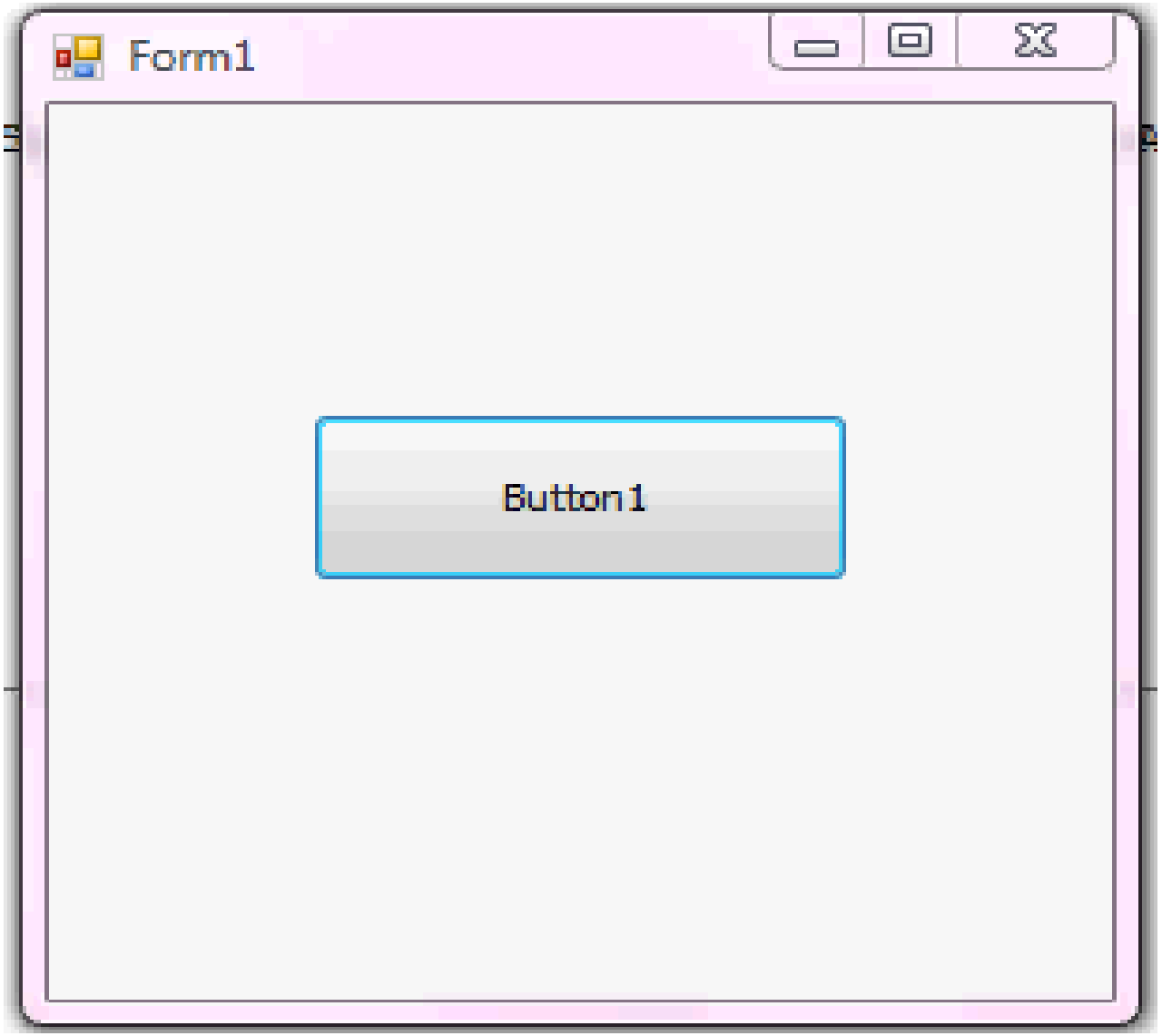
مثال

❖ضبط درجة شفافية النموذج حيث يخبو حتى يختفي تماماً عند الضغط على زر b1 ....

كود b1

```
Dim sngopacity As single
For sngopacity = 1 To  0  Step  -0.05
        Me.opacity= sngopacity
        Me.refresh (  )                  //لجعل النموذج يعيد رسم نفسه
        system.Threading.Thread.Sleep(200)      // توقف قليلاً
Next
Me.opacity=1                    // لعرض النموذج مرة أخرى
```

❖العداد نوعه Single ليأخذ قيماً صحيحة وغير صحيحة .
❖جملة Sleep تجعل الفيجوال ينتظر قليلاً بين كل دورة وأخرى
(جزء من الألف من الثانية ٢٠٠ ميللي ثانية)


Form1
Button1

تكرار الأوامر لعدد غير محدد من المرات Do .... Loop

❖إستمرار التكرار حتى يتحقق شرط معين

أشكال Do . . . Loop

١

```
Do
        [أوامر]
Loop
```

❖هذا الشكل يؤدي إلى تكرار لا نهائي

٢

```
Do
        [أوامر]
        if <expression> Then  Exit Do
Loop
```

❖يستمر التكرار حتى يصبح التعبير صوابا وهذا التعبير مبني على متغير ما يتم تعديل قيمته داخل التكرار

مثال

```
X=1
Do
        Debug.writeline(x)
        If x=5 then exit do
        x=x+2
Loop
```

```
X=1
Do
        x=x+1
        Debug.writeline(x)
        If x=5 then exit do
Loop
```

أشكال Do . . . Loop

- Do While expression
  [أوامر]
  Loop

٣

❖يستمر التكرار طالما التعبير صواباً

Do until expression
[أوامر]

Loop

٤

❖يستمر التكرار حتى يصبح التعبير صوابا (أي طالما التعبير خاطئ)

❖في الشكلين (٣و ٤) يتم تقييم التعبير قبل الدخول للحلقة التكرارية

مثال

```
X=1
Do  while  x<>5
          x=x+1
          Debug.writeline(x)
Loop
```

النتيجة:

2
3
4
5

```
X=1
Do until  x=5
          x=x+1
          Debug.writeline(x)
Loop
```

أشكال Do . . . Loop

٥

```
Do
        [أوامر]
Loop While expression
```

٦

```
Do
        [أوامر]
Loop until expression
```

❖في الشكلين (٥و ٦) يتم تقييم التعبير مؤخراً .
أي سيتم تنفيذ التكرار مرة واحدة قبل تقييم التعبير

مثال

❖طباعة أول عشر أرقام تقبل القسمة على ٣ ...

```
Dim  intseek As integer =1
Dim intfound As integer =0
Do until intfound=10
        if intseek mod 3 = 0 Then
                debug.writeline (intseek )
                intfound=intfound +1
        End If
        intseek = intseek + 1
Loop
```

❖ **intseek** لحفظ الأرقام المسلسلة التي سيتم إختبارها للتعرف ما إذا كانت تقبل القسمة على ٣ أم لا .

❖ **intfound** عداد التكرار يزيد بمقدار واحد في كل مرة يتم إيجاد رقماً يقبل القسمة على ٣.

النتيجة

- طباعة أول عشر أرقام تقبل القسمة على ٣ ...

3
6
9
12
15
18
21
24
27
30