

Iteration -- Once Is Not Enough

**FIT
100**

Though people don't like to repeat themselves, repetition is a valuable facility that a computer can provide. If program instructions are to be performed more than once, as in Alphabetize CDs, repetition is needed

© Copyright University of Washington 1999-2000

FIT 100 Two Additional Control Statements

- ❖ The conditional statement (If-Then-Else) is the only way (so far) to *control* which statements are executed
- ❖ We will introduce two more:
 - ❑ Elseif -- a variation on the If-Then-Else for long sequences of tests
 - ❑ Do While -- a control facility allowing statements to be repeated as long as some condition is true

Programming languages have other control statements, but these are enough to do any programming

© Copyright University of Washington 1999-2000

FIT 100 Elseif

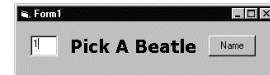
- ❖ Elseif solves the problem of testing a long sequence of alternatives

```
If <T/F condition> Then
  <statement list>           Stmts for 1st cond
Elseif <T/F condition> Then
  <statement list>           Stmts for 2nd cond
Elseif <T/F condition> Then
  <statement list>           Stmts for 3rd cond
Elseif <T/F condition> Then
  <statement list>           Stmts for 4th cond
...
Else
  <statement list>           Stmts for otherwise
Endif
```

© Copyright University of Washington 1999-2000

FIT 100 Example

```
If txtNum.Text = 1 Then      Executed if Text = 1
  MsgBox("John")
Elseif txtNum.Text = 2 Then  Executed if Text = 1 and Text = 2
  MsgBox("Paul")
Elseif txtNum.Text = 3 Then  Executed if Text = 1 or 2 and Text = 3
  MsgBox("George")
Elseif txtNum.Text = 4 Then  Executed if Text = 1 or 2 or 3 and Text = 4
  MsgBox("Ringo")
Else
  MsgBox("Who?")
Endif
```



© Copyright University of Washington 1999-2000

FIT 100 Mini-Exercise #1

What does this print?

```
Dim x As Integer
x = 10
If x = 1 Then
  Print "octopus"
Elseif x = 2 Then
  Print "squid"
Else
  Print "clam"
Endif
Print "mollusc"
```

© Copyright University of Washington 1999-2000

FIT 100 Mini-Exercise #1 -- Answer

What does this print?

```
Dim x As Integer
x = 10
If x = 1 Then
  Print "octopus"
Elseif x = 2 Then
  Print "squid"
Else
  Print "clam"
Endif
Print "mollusc"
```

clam
mollusc

© Copyright University of Washington 1999-2000

FIT 100

Contrast Elseif With Nested If

- ❖ Elseif is not a nested test as seen before, though it is similar

```

If txtNum.Text = 1 Then
  MsgBox("John")
Elseif txtNum.Text = 2 Then
  MsgBox("Paul")
Elseif txtNum.Text = 3 Then
  MsgBox("George")
Elseif txtNum.Text = 4 Then
  MsgBox("Ringo")
Else
  MsgBox("Who?")
End If

If txtNum.Text = 1 Then
  MsgBox("John")
Else
  If txtNum.Text = 2 Then
    MsgBox("Paul")
  Else
    If txtNum.Text = 3 Then
      MsgBox("George")
    Else
      If txtNum.Text = 4 Then
        MsgBox("Ringo")
      Else
        MsgBox("Who?")
      End If
    End If
  End If
End If

```

FIT 100

Caution With Else If

- ❖ An If statement that uses Else If passes through all of the previous cases before reaching a given test ... think about the consequences

```

If someVar < 20 Then
  MsgBox("Less than 20")
Elseif someVar < 10 Then
  MsgBox("Less than 10")
Else
  ...
Endif

```

Will this MsgBox ever be executed?

FIT 100

Repeating Terms

- ❖ Iteration is the repeated execution of a series of statements in programming
- ❖ To perform iteration, programming languages include special statements often called *iteration statements*
- ❖ There are two crucial components of all iterations:
 - The statements that will be repeated -- called the loop body
 - A test specifying when to repetition stops -- termination test
- ❖ Additionally, loops typically have at least one variable that is explicitly changed "inside" the loop -- this is called the iteration variable

Some value *must* change between consecutive iterations, or else the loop will never terminate ... it is an infinite loop

FIT 100

General Form Of VB6 Iteration

- ❖ VB6, like most languages, has several iteration statements. We'll just use one in CSE/IMT 100:

```

Do While <termination condition>
  <statements>
Loop

```

- ❖ The semantics are as follows:
 - The termination condition is tested and if it is false the statements are all skipped; execution continues after Loop
 - If it is true, the statements are performed once
 - The termination condition is tested again, and if it is false the loop is over and the statements are skipped; continue after Loop
 - If it is true, the statements are performed a second time
 - ...

FIT 100

An Example

- ❖ An easy way to get the idea of iteration is to print out the iteration variable ...

```

Option Explicit

Private Sub Form_Click()
  Dim iterateVar As Integer
  iterateVar = 0
  Do While iterateVar < 10
    iterateVar = iterateVar + 1
    Print ("iterateVar is" & iterateVar)
  Loop
End Sub

```

Declaration of iteration variable

Initialization of iteration variable

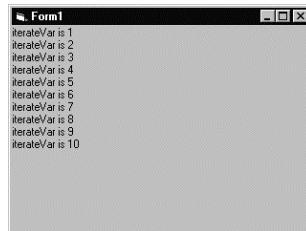
Termination Condition

Increment of the iteration variable

Loop Body

FIT 100

Execution of Example



- ❖ Try the same computation with a different termination condition

FIT 100 Mini-Exercise #2

❖ What does this code print?

```
Dim i As Integer
i = 2
Do While i <= 4
  Print i
  i = i + 1
loop
```

FIT 100 Mini-Exercise #2 -- Answer

❖ What does this code print?

```
Dim i As Integer
i = 2
Do While i <= 4
  Print i
  i = i + 1
loop
```

2
3
4

FIT 100 Mini-Exercise #3

❖ What does this code print?

```
Dim i As Integer
i = 10
Do While i <= 4
  Print i
  i = i + 1
loop
```

FIT 100 Mini-Exercise #3 -- Answer

❖ What does this code print?

```
Dim i As Integer
i = 10
Do While i <= 4
  Print i
  i = i + 1
loop
```

It doesn't print anything!

FIT 100 Mini-Exercise #4

❖ What does this code print?

```
Dim i As Integer
i = 1
Do While i >= 1
  Print i
  i = i + 1
loop
```

FIT 100 Mini-Exercise #4 -- Answer

❖ What does this code print?

```
Dim i As Integer
i = 1
Do While i >= 1
  Print i
  i = i + 1
loop
```

1
2
3
4
5
6
.....

This is an infinite loop!

**FIT
100**

To get out of an infinite loop:

In Visual Basic, type
control-break

Then you are in the debugger, and can see where
you are stuck. Select "continue" to proceed

If a program (any program) gets stuck in Windows, you
can also type control-alt-delete and get a task
manager. Select the offending task and end it.
Caution: you'll lose any changes for that program
since the last save.

© Copyright University of Washington 1999-2000

**FIT
100**

Mini-Exercise #5

- ❖ Suppose we have a procedure "squid" that takes a
single integer argument. The argument is only read
by squid, not changed. Write a loop that calls a
"squid" with 2, 4, 6, 8, 10.

© Copyright University of Washington 1999-2000

**FIT
100**

Mini-Exercise #5 -- Answer

- ❖ Suppose we have a procedure "squid" that takes a
single integer argument. The argument is only read
by squid, not changed. Write a loop that calls a
"squid" with 2, 4, 6, 8, 10.

```
Dim i As Integer
i = 2
Do While i <= 10
  Call squid(i)
  i = i + 2
loop
```

© Copyright University of Washington 1999-2000