(If … Then) Statement:

This is the simplest control structure which ask the computer to perform a certain action specified by the VB expression if the condition is true. However, when the condition is false, no action will be performed. The general format for the (**If - Then**) statement is:

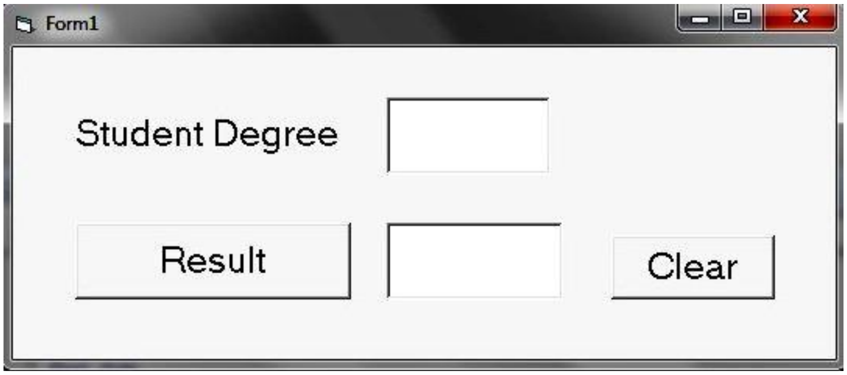
```
If condition Then

VB expression

 End If
```

Example1: Design a form with two text boxes and two Command buttons. Write a code so when run project and click on command1 **(Result)** display the word "**Pass**" if the student degree equal or more than **50**, and when click on Command2 clear the text boxes.

Solution:



```
Private Sub Command1_Click()

If Val(Text1.Text) >= 50 Then

Text2.Text = "Pass"
End If

End Sub
```

```
Private Sub Command2_Click()
Text1.Text = ""
Text2.Text = ""
End Sub
```

(If … Then … else) Statement:

The **If – Then - Else** statement allows the programmer to specify that different actions to be performed when a certain action specified by the VB expression if the condition is True than when the condition is false, an alternative action will be executed. The general format for the **(If - Then – Else)** statement is:

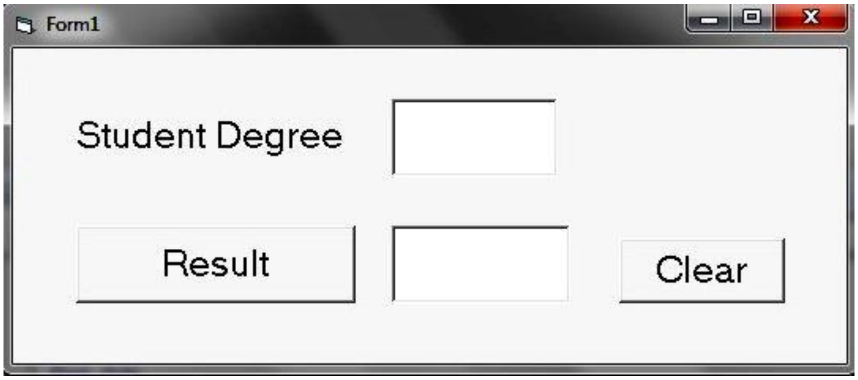
```
If condition Then

VB expression
Else
VB expression

End If
```

Example2: Design a form with two text boxes and two Command buttons. Write a code so when run project and click on command1 **(Result)** display the word "**Pass**" if the student degree equal or more than **50**, and display "**Fail**" if the student degree less than 50.

Solution:



```
Private Sub Command1_Click()

If val(Text1.Text) >= 50 Then
 Text2.Text = "Pass"
Else
Text2.Text = "Fail" End If
End Sub
```

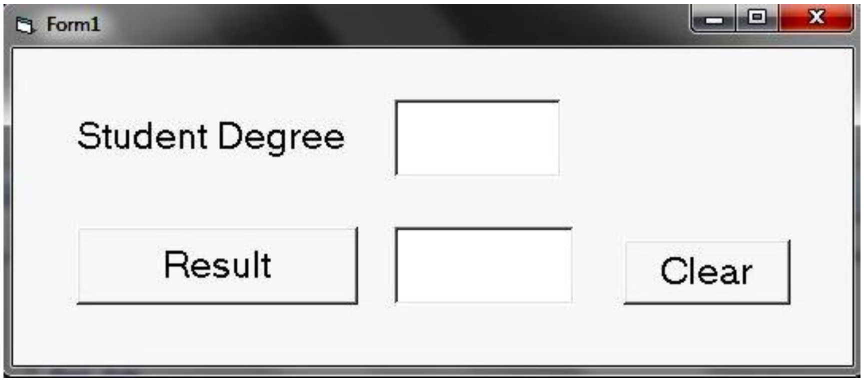
```
Private Sub Command2_Click()
Text1.Text = ""
Text2.Text = ""
 End Sub
```

Operator	Description	Example
<	Less than operator	if 19 < 20 Then
>	Greater than operator	if 20 > 19 Then
=	Equal to operator	if a = b Then
<>	Not equal to operator	if a <> b Then
<=	Less than or equal to operator	if 19 <= b Then
>=	Greater than or equal to operator	if 19 >= b Then

Operator	Description
And	And Operator - If both conditions are true, then result is true
Or	Or Operator - If either of the conditions is true the result will be true
Not	Not Operator - If the condition is false then the result will be true, or vice versa.

Example3: Design a form with two text boxes and two Command buttons. Write a code so when run project and click on command1 **(Result)** display the word "**Pass**" if the student degree between **50-100**, and display "**Fail**" if the student degree between **0-49**, and display "**Wrong Input**" if the degree less than **0** or more than **100**.

Solution:



```
Private Sub Command1_Click()
If val(Text1.Text) >= 50  And  val(Text1.Text) <= 100 Then
 Text2.Text = "Pass"
Else
If val(Text1.Text) < 0 Or  val(Text1.Text) > 100 Then
Text2.Text = "Wrong Input"
End If
 End If
If val(Text1.Text) >= 0  And  val(Text1.Text) < 50 Then
Text2.Text = "Fail"
End If
End Sub
```

```
Private Sub Command2_Click()
Text1.Text = ""
Text2.Text = ""
 End Sub
```

"Declare In Memory" (Dim) method:

Dim is a statement which is used to declare the variables in Visual Basic.

The general formula for the definition of variables:

DIM variable name AS data type.

Example4: Repeat example 3 using **Dim method.**

Solution:

```
Dim A As Integer
 Dim B As String
```

```
Private Sub Command1_Click()
A = Val(Text1.Text)
If A >= 50 And A <= 100 Then
 B = "Pass"
Else
If A < 0 Or A > 100 Then
 B = "Wrong Input"
End If
End If
If A >= 0 And A < 50Then
```

B = "Fail"

End If

Text2.Text = B

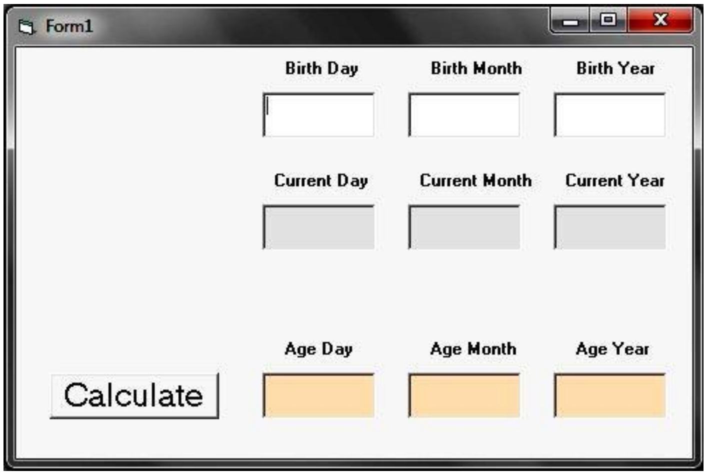
End Sub

Integer: includes a whole numbers from -32768 to 32767

String: includes a collection of letters such as "Hello". This is used to store words and sentences.

Example5: design a simple program to calculate approximately your age (dd/mm/yyyy).

Solution:



Code:

Dim d, m, y, d1, m1, y1 As Integer

Private Sub Command1_Click()

d = Val(Text1.Text)

```
m = Val(Text2.Text)
y = Val(Text3.Text)
d1 = Val(Text4.Text)
m1 = Val(Text5.Text)
y1 = Val(Text6.Text)

Text7.Text = d1 - d
Text8.Text = m1 - m
Text9.Text = y1 – y
 End Sub
```