

Control Statements

If

If you can keep your head when all about you
Are losing theirs and blaming it on you,
If you can trust yourself when all men doubt you
But make allowance for their doubting too,
If you can wait and not be tired by waiting,
Or being lied about, don't deal in lies,
Or being hated, don't give way to hating,
And yet don't look too good, nor talk too wise:
THEN

Yours is the Earth and everything that's in it,
And--which is more--you'll be a Man, my son!

--*Rudyard Kipling*



The Boolean Variable

named after **George Boole**

Famous 19th Century British Mathematician

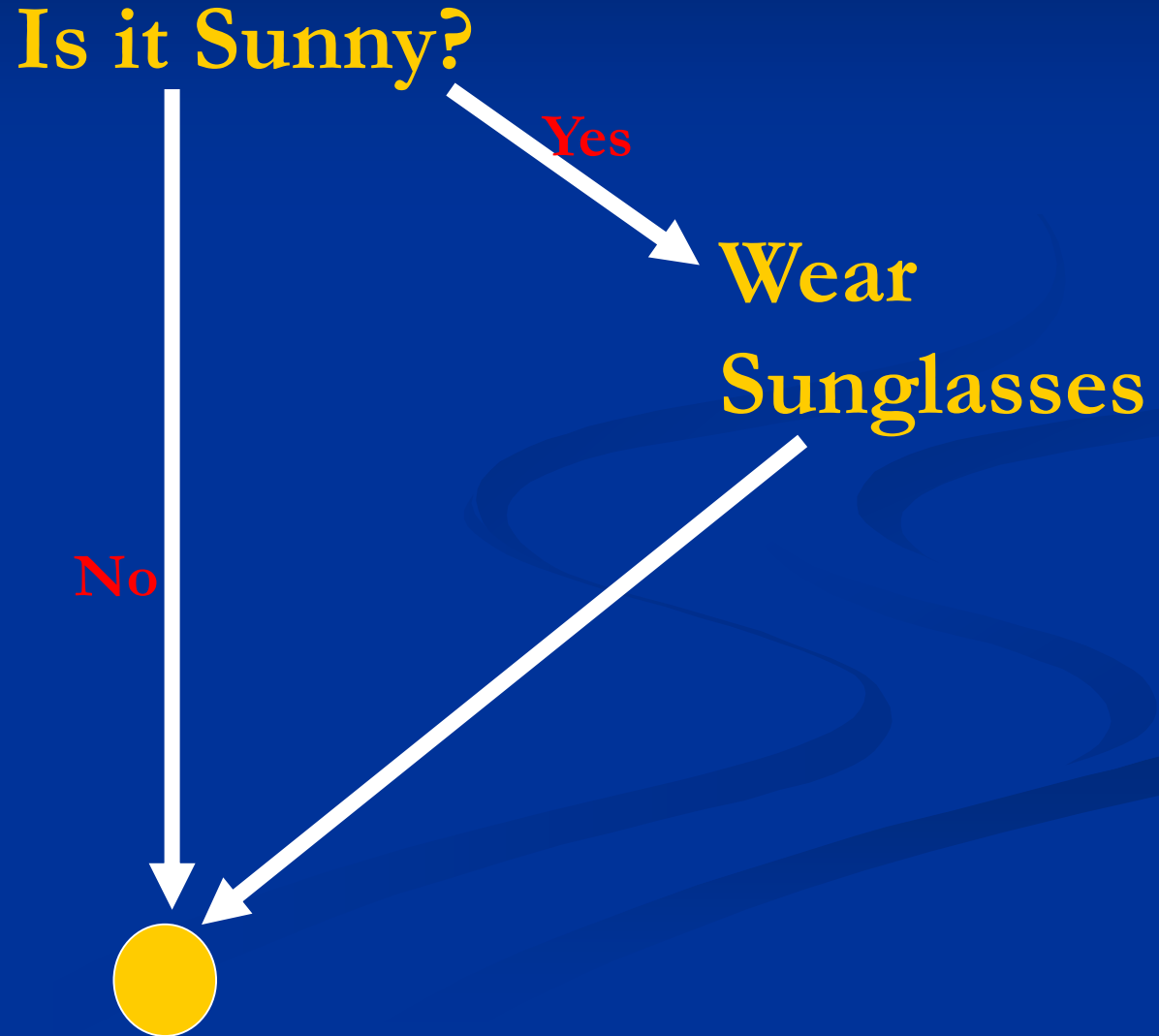
A boolean variable can have only one of two values;

TRUE or FALSE

Boolean Example

This simple question can be used to represent a boolean expression in Visual Basic. It is either true or it is false.

Based on the answer the flow of the program is either diverted to the sunglasses or it is not.



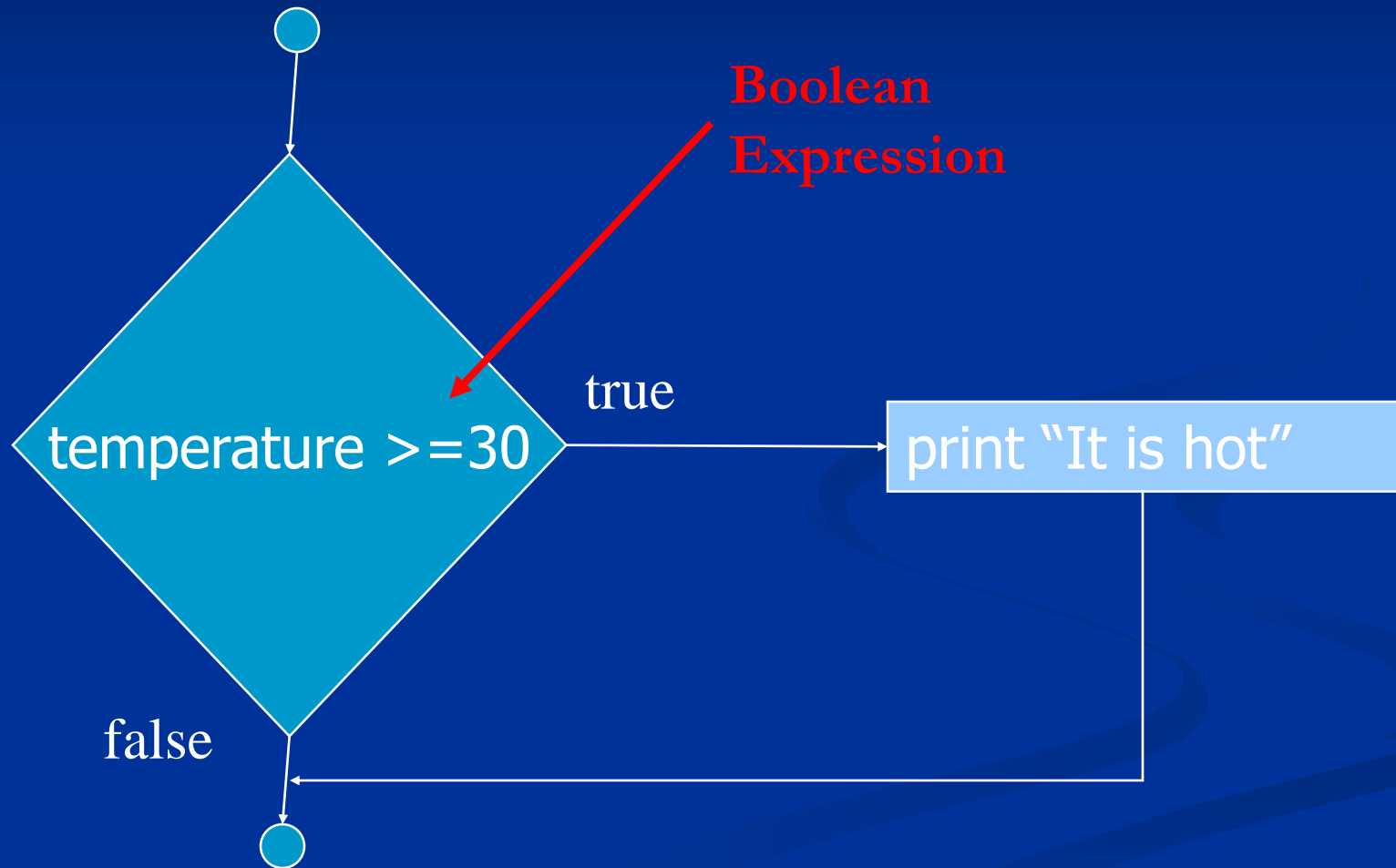
Boolean Variables and Expressions

The 'if statement' tell us what should occur, (what actions or what consequences) IF the result of the boolean variable or expression is false.

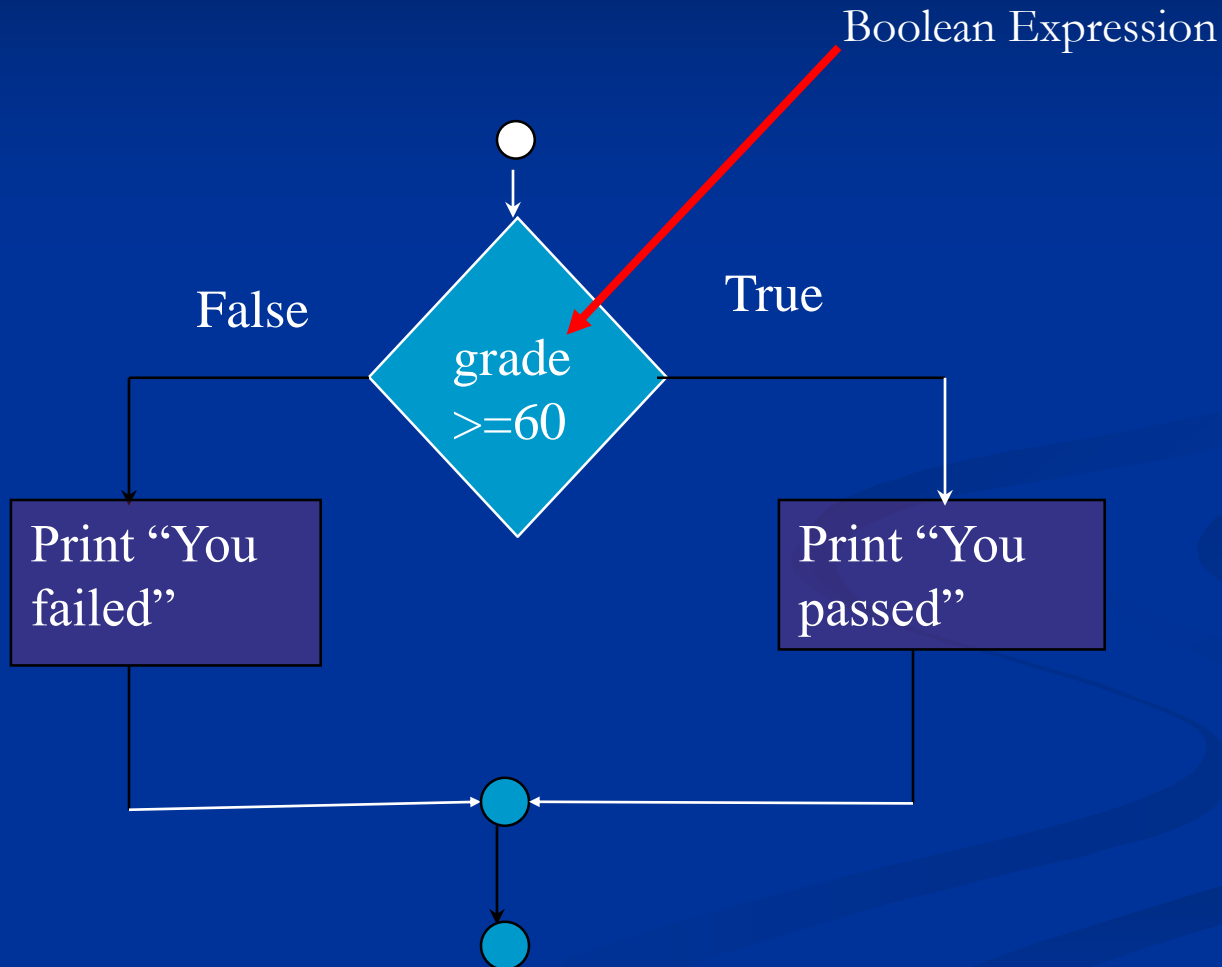
Instead, various operators are used to compare values. Based on the result of these comparisons, the expression is determined to be true or false.

Lets build a boolean expression within an 'if-statement' based on the question posed on the next slide.

if Flow Chart



if/else Flow Chart



A Simple If Else Statement

```
Private Sub btnClick_Click(ByVal sender
    Dim mark As Integer
    mark = Val(txtMark.Text)

    If (mark >= 50) Then
        lblMark.Text = "Pass"
    Else
        lblMark.Text = "Fail"
    End If

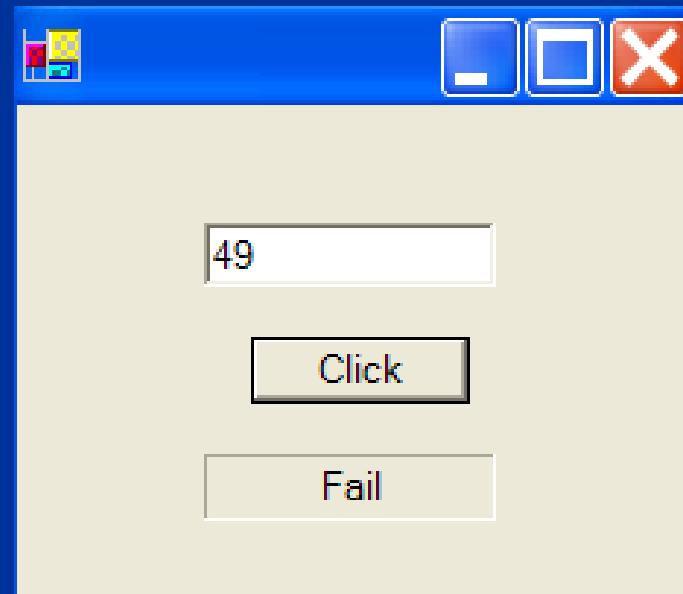
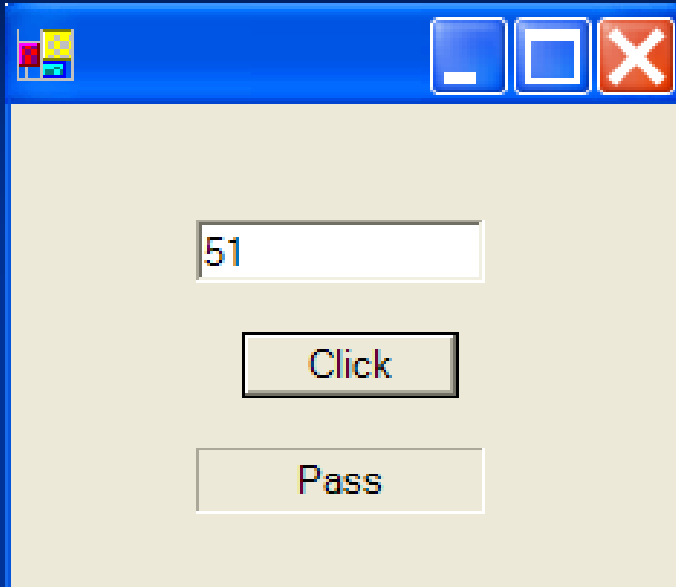
End Sub
```

The contents of the brackets in an if statement basically ask a question that can only be answered true or false. In this case the question is, "Is the value of mark greater than (or equal to) 50 or not?"

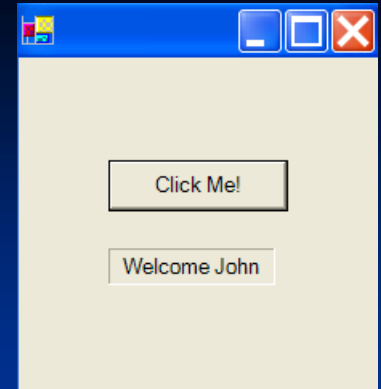
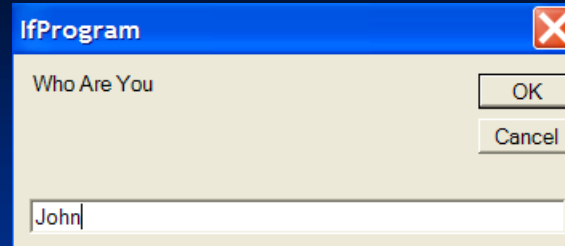
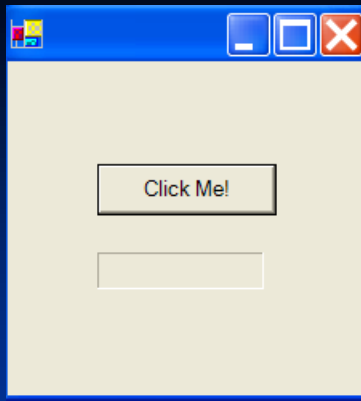
If the answer is yes, the code on the next line is run.

If value of mark is not greater than (or equal to) 50 then the contents of the brackets evaluates to false. This causes the code beneath the if statement to be skipped and only the code after the else statement to be run.

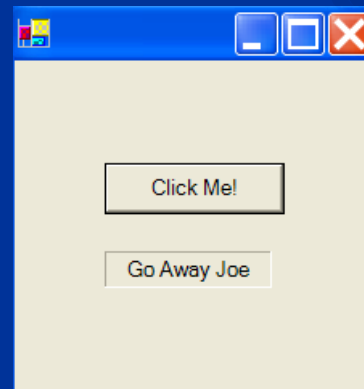
Running The If/Else Program



Write A Password Program



Write a program that prompts the user to enter a name. This name will then be compared With a String variable you have chosen within the program. If the words match allow them In with a greeting that calls them by name. If the names do match up, tell the user to go away Using the name they supplied.



A Solution

```
Private Sub btnClick_Click(ByVal sender As System.  
    Dim name As String  
    name = InputBox("Who Are You")  
  
    If (name = "John") Then  
        lblMark.Text = "Welcome " & name  
    Else  
        lblMark.Text = "Go Away " & name  
    End If  
  
End Sub
```

if / else if / else

if statements allow you to take in account 1 set of circumstances.

if/else statements allow you take into account 2 sets of circumstances.

if/else if/else statement can be strung together indefinitely to compare many sets of circumstances.

Pseudo-Code

if

...I was rich, I mean really, really rich I could buy myself a Rolls-Royce with all the options.



Or...

...or

else if

...I was only a little rich I might have to get something a little more inexpensive like a Mercedes.



Or...

...or

else if

...I was just wealthy I could drive away in a 'Hummer'



Or...

...or

else if

...I was earning a decent salary I might splurge for a Cadillac



Or...

...or

else if

...I was only making an average salary I would probably only be able to afford a Chevy Impala.



Or...

...or

else if

...I was Mr. Wright



Is this dog showing his disgust for the car in a way that only dogs can?

Or...

...or

else

I would have to settle for getting around by.....



Comparative Operators

Operator	Use
=	Tests whether two values are equal.
<>	Tests whether two values are <i>not</i> equal.
<	Tests whether the first value is less than the second value.
>	Tests whether the first value is greater than the second value.
<=	Tests whether the first value is less than or equal to the second value.
>=	Tests whether the first value is greater than or equal to the second value.

The Report Card

Write a program which prompts the user for a mark through an input box. The program should contain a button called btnGrade which will evaluate the mark output a letter grade based on the mark.

Use the following:

≥ 90	A+
≥ 80	A
≥ 70	B
≥ 60	C
≥ 50	D
< 50	F

Report Card Grade

```
Private Sub btnGrade_Click(ByVal sender As
    Dim mark As Integer
    mark = Val(InputBox("Enter A Mark"))

    If (mark >= 90) Then
        MsgBox("A+")
    ElseIf (mark >= 80) Then
        MsgBox("A")
    ElseIf (mark >= 70) Then
        MsgBox("B")
    ElseIf (mark >= 60) Then
        MsgBox("C")
    ElseIf (mark >= 50) Then
        MsgBox("D")
    Else
        MsgBox("F")
    End If
End Sub
```

No need to ask a comparative question on last statement. All the possibilities have been used, the only outcome is 'F'.