

Mr. Giansante



Visual Basic

If - Then Statements

August 2016

Decisions Using If - Then - Else

In general, **If-Then** statements fall into two categories.

1. There is only one action to be taken
2. There are multiple actions to be taken

One Action to be Taken

If there is only one action to be taken, the If-Then statement can be one line.

```
If (condition) then (your code goes here)
```

Example:

```
If (tem <= 0) then Label1.Text = "Freezing"
```

Multiple Actions to be Taken

If there are multiple actions to be taken, the If-Then statement must be on more than one line and contain an **End If**.

```
If (condition) then
  (your code goes here)
  (your code goes here)
  (your code goes here)
End If
```

Example:

```
If user="John Smith" then
  Label1.Text = "Phone: 4792-0000"
  Label2.Text = "City: Buenos Aires"
  Label3.Text = "Country: Argentina"
End If
```

If-Then-Else

```
If (condition) then
  (your code goes here)
Else
  (your code goes here)
End If
```

Example:

```
If temperature <= 0 then
  Label1.Text = "Freezing or below."
Else
  Label1.Text = "Above Freezing."
End If
```

Comparators

The following comparators can be used in the condition part of an If-Then statement.

<	less than
>	greater than
=	equal to
<=	less than or equal to
>=	greater than or equal to
<>	not equal to

Boolean Operators

The following Boolean operators can be used in the condition part of an If-Then statement.

And
Or
Not

Example: Is a number even or odd?

Use the `Mod()` operator which returns the remainder from division. If you divide a number by 2 and there is no remainder, then the number is even.

```
If x mod 2 = 0 then
  MessageBox.Show("Number is even")
Else
  MessageBox.Show("Number is odd")
End if
```

If - Then - Else Examples

Example 1

Let's say we ask the user to type a number between 10 and 20 in a TextBox Control (named TextBox1) and we want to display a message if they do something wrong ...

First, we could check if what they typed in the TextBox Control is actually a number ...

```
If IsNumeric(TextBox1.Text) = False Then
    MsgBox.Show("Please enter numbers only.")
    TextBox1.Clear()
    TextBox1.Focus()
Exit Sub
End If
```

Next, we could check if the number they typed is greater than or equal to 10 ...

```
If Val(TextBox1.Text) < 10 Then
    MsgBox.Show("You must enter a number greater than or equal to 10")
    TextBox1.Clear()
    TextBox1.Focus()
Exit Sub
End If
```

Next, we could check if the number they typed is less than or equal to 20 ...

```
If Val(TextBox1.Text) > 20 Then
    MsgBox.Show("You must enter a number smaller than or equal to 20")
    TextBox1.Clear()
    TextBox1.Focus()
Exit Sub
End If
```

Example 2

Let's say you ask the user to type in a password in a TextBox Control (named TextBox1) and you want to display a MsgBox telling them if the password they typed is valid. Let's assume the valid password is "welcome".

```
If TextBox1.Text = "welcome" then
    MsgBox.Show("Correct password.")
Else
    MsgBox.Show("Wrong password.")
End If
```

If we wanted to make the above example a bit better by allowing the user to type the password in uppercase or lowercase letters, we could do the following ...

```
If LCase(TextBox1.Text) = "welcome" then
    MsgBox.Show("Correct password.")
Else
    MsgBox.Show("Wrong password.")
End If
```

Let's say that we wanted to write code where the user would have to type a Login (in a TextBox Control named TextBox1) and a Password (in a TextBox Control named TextBox2). Let's assume the valid login is "user" and the valid password is "welcome".

```
If LCase(TextBox1.Text) = "user" And LCase(TextBox2.Text) = "welcome" then
    MsgBox.Show("Correct login and password.")
Else
    MsgBox.Show("Wrong login or password.")
End If
```

If - Then Exercises - Part I

Name

Date

Write the code to examine an integer variable named "x" and put the word "Negative" in a Label Control (named Label1) if the number is less than zero, otherwise put the word "Positive".

```
If _____ Then
    Label1. _____ = _____
Else
    Label1. _____ = _____
End If
```

Write the code to examine an integer variable named "x" and put the word "Even" in a Label Control (named Label1) if the number is even, otherwise put the word "Odd".

Hint: Use the **Mod()** function to see if there is a remainder when you divide by 2.

```
If _____ Then
    Label1. _____ = _____
Else
    Label1. _____ = _____
End If
```

Write the code to examine an integer variable named "x" and put the word "Divisible by 5" in a Label Control (named Label1) if the number is divisible by 5, otherwise put the word "Not Divisible by 5".

Hint: Use the **Mod()** function to see if there is a remainder when you divide by 5.

```
If _____ Then
    Label1. _____ = _____
Else
    Label1. _____ = _____
End If
```

Write the code to check if a variable named "Mark" is less than zero or greater than one hundred, and if it is, the phrase "Mark out of range" should appear in a Label control (named Label1).

```
If _____ Then
    Label1. _____ = _____
End If
```

If - Then Exercises - Part II

Name

Date

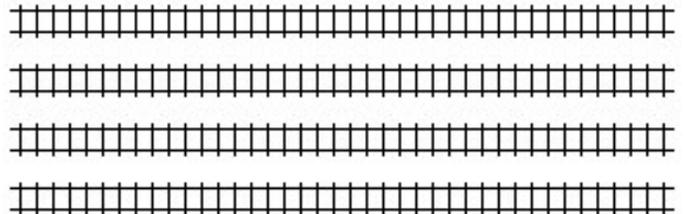
Assuming you have a string variable named **LastName**. Write the code to accomplish the following:

a. Put the words "Name too short." in a Label Control (named Label1) if the name is less than three characters long.

b. Put the words "Hyphenated Name." in a Label Control (named Label1) if the customer's last name has a hyphen in it.
Hint: Use the **Instr()** command to search for the "-" character.

Train Track Safety System

On a bridge, four train tracks run parallel to each other. Sensors on each track determine which tracks have trains on them and set four different Boolean variables accordingly. The variables are named: track1, track2, track3 and track4. Recall that a Boolean variable can have one of two values: True or False. For example: if there is a train on track 1 then track1 is set to "True", otherwise it is set to "False".



Situation 1 ... Let's say the tracks are set up so closely that two trains cannot be on adjacent tracks without hitting each other. Write the code to display a MessageBox if trains are on adjacent tracks.

Situation 2 ... Let's say that the bridge has a weight restriction so that a maximum of three trains can be on the bridge at once. Write the code to display a MessageBox if there are four trains on the tracks.

Situation 3 ... Let's say that the bridge has a weight restriction so that a maximum of two trains can be on the bridge at once. Write the code to display a MessageBox if there are three or more trains on the tracks.